



UNIVERSIDAD NACIONAL DE ROSARIO

TESINA DE GRADO
PARA LA OBTENCIÓN DEL GRADO DE
LICENCIADO EN CIENCIAS DE LA COMPUTACIÓN

Visión por computadora en la Predicción de la
Enfermedad de Alzheimer en Pacientes Mayores
por Medio de la Evaluación del Rendimiento en
Actividades Diarias.

Autor:
Álvaro Gómez Uría

Director:
Dr. François Brémond

Departamento de Ciencias de la Computación
Facultad de Ciencias Exactas, Ingeniería y Agrimensura
Av. Pellegrini 250, Rosario, Santa Fe, Argentina

23 de septiembre de 2015

Resumen

La enfermedad de Alzheimer (EA) es una de las demencias más comunes que afecta a gran parte de la población de la tercera edad. Un diagnóstico oportuno permitiría a los médicos actuar de manera preventiva en el cuidado de los pacientes y de esa manera poder mejorar la calidad de vida extendiendo su independencia en las actividades rutinarias. No obstante, un correcto diagnóstico dentro de las demencias existentes es una tarea dificultosa debido a que sus síntomas suelen confundirse con otros síndromes, como ser el caso del Deterioro Cognitivo Leve (*Mild Cognitive Impairment* - MCI), o pueden darse casos donde el comportamiento es completamente impredecible.

Los métodos actuales de evaluación implican escalas psicológicas y puntuaciones que pueden ser bastante subjetivas o parciales, ya que por ejemplo, dependen de la opinión de la persona a cargo del paciente para recopilar datos sobre el desempeño en sus funciones diarias. En términos científicos podríamos denominar a esto como “sesgo de investigación”.

Por lo tanto, se propone un método para un diagnóstico temprano de la enfermedad de Alzheimer, que sirva de apoyo para los médicos. Para ello, se evalúa la autonomía de los pacientes por medio de su desempeño en las actividades de la vida diaria teniendo en cuenta sus comportamientos (o rendimientos) en las mismas, a través de una representación jerárquica construida a partir de las actividades reconocidas por un sistema de monitoreo de la actividad basado en visión. Para enriquecer tales perfiles también se incorporaron estándares básicos de personas con respecto a su andar durante la realización de tareas físicas. Además se lo compara con técnicas conocidas.

Palabras claves

Visión por computador, detector de personas, seguimiento de objetos por trayectoria, feet detection, reconocimiento de actividades, aprendizaje automático, clasificación, ontología basada en restricciones, demencia, Alzheimer, autonomía.

Abstact

Alzheimer's disease (AD) is the most common type of dementia which affects a large part of the elder population. An early diagnosis would allow specialist to act preventively in the care of patients and thus can improve the quality of life extending their independence in activities of daily living. However, a correct diagnosis is a difficult task because their symptoms are often confused with other syndromes, such as the case of Mild Cognitive Impairment (MCI) or any other cases where the behavior is totally unpredictable.

The current methods for the assessment involves psychological scales and rates that have personal biases, i.e. they depend on the opinion of the caregiver.

Therefore, the proposed system may become a very useful tool that provide clinicians a diagnostic with relevant information and improve autonomy assessment in AD or MCI patients in real time decreasing observer bias. The autonomy of patients is evaluated through their performance in activities of daily living. To enrich such profiles, gait parameters are also incorporated during the performance of physical tasks.

Keywords

Computer Vision, people detection, tracking, feet detection, activity recognition, machine learning, classification, constraint-based ontology, dementia, Alzheimer, autonomy.

Agradecimientos

A las primeras personas a las que les quiero agradecer son sin dudas a mi familia (Mamá, Papá, hermanos, sobrinitas y abuelos). A pesar que después de tanto tiempo siguen sin entender algo de lo que hice y hago, el apoyo constante en las buenas y en las malas ha sido fundamental para llegar al final. La emoción de ellos en cada llamado después de un final para comunicarles el tan ansiado “Aprobé” era sin dudas la mejor parte de la historia.

Por debajo de mi familia se encuentran mis compañeros, a esta altura amigos, que compartieron todo este largo camino conmigo (aunque algunas se hayan quedado en el medio) y siempre han estado presentes para esas largas jornadas de estudio y para ayudarme con mis constantes dudas en las previas de los exámenes. Ellos son: Lichu, Gonza, Emilio, Pata, Clemen, Nacho, Wanda, Anto, Gery, Ore, Lucio, Peto, Anra, Zeta, Itox y Rocko.

A mi amigo, ex-compañero de equipo de trabajo y “co-director” de esta tesina Pablo Pusiol. Fue la persona justa para tener cerca en mi estadía en Francia y luego el consejero ideal para este trabajo.

También quiero agradecerle a mi director, François Brémond, por abrirme las puertas de STARS Team y confiar en mí para este trabajo, y a la persona que me guió durante toda mi pasantía, Post-doc Carlos Fernando Crispim-Junior.

Por último, a todos mis profesores y en especial a Ana Casali y Raúl Kantor, dos personas que hicieron y hacen mucho por esta carrera, y siempre están muy cerca del alumno para facilitarnos las cosas. Sin dudas sin ellos no habría carrera.

Índice general

	Página
Resumen	III
Abstract	IV
Agradecimientos	V
Índice general	VI
Índice de figuras	VIII
1 Introducción	1
1.1. El problema a abordar	1
1.2. Enfermedad de Alzheimer y otras demencias seniles	2
Estado del arte sobre la clasificación y evaluación	3
1.3. Propuesta	4
Visión por Computadora	5
Aprendizaje automático	5
Contribuciones	5
1.4. Organización del trabajo	6
2 Detección de Pacientes en Escena	7
2.1. Detección de Personas en 3D	8
Algoritmos de Substracción de Fondos	8
Algoritmo propuesto	10
Modelado de la alta varianza en la medición de la profundidad	10
Modelo del <i>background</i>	10
Clasificación de los valores de profundidad	11
Eliminación del Ruido	12
Resultados del algoritmo propuesto	12
2.2. Seguimiento por Trayectoria	13
Proceso del seguimiento de objetos	13
Evaluación de la detección	14
Corrección de la detección	14
Tracker de apariencia discriminatorio	14

Tracker KLT	15
Filtración del ruido	16
2.3. Feet Detection	16
Detección del suelo	17
Detección de la piernas y de la posición de los pies	17
Obtención de parámetros de la marcha	17
3 Reconocimiento de Actividades	19
3.1. Eventos	19
3.2. Composición de un modelo de eventos	20
3.3. Restricciones	21
3.4. Ejemplos de modelos	22
Modelo de un evento complejo	22
Modelos de nuestros eventos	23
4 Clasificación de los pacientes	27
4.1. Protocolo médico	28
Actividades guiadas	28
Actividades semi-guiadas	29
4.2. Representación del comportamiento	30
Adición de características derivadas	30
Lista completa de características	32
4.3. Selección de características	33
4.4. Algoritmos de Machine Learning	34
Aprendizaje supervisado	34
4.5. Clasificadores	35
Clasificador <i>Naïve Bayes</i>	35
4.6. Evaluación de los algoritmos	38
5 Sistema para la predicción de demencia	41
5.1. Sistema de Monitoreo de Eventos	42
5.2. Acerca los pacientes	42
5.3. Evaluación de la autonomía y predicción de demencia	43
Selección de características. Wrapper en Weka	44
Uso del clasificador <i>Naïve Bayes</i>	45
5.4. Validación del sistema	46
5.5. Resultados	47
Predicción del diagnóstico de demencia	48
Actividades guiadas	49
Actividades semi-guiadas	50
Todas las actividades	51
Predicción de la autonomía	52
Actividades guiadas	53
Actividades semi-guiadas	54

Todas las actividades	55
Distintas técnicas validación y entrenamiento	56
6 Conclusión	57
6.1. Discusión	57
6.2. Contribuciones a futuro	58
Bibliografía	59

Índice de figuras

1.1. Diagrama central del sistema.	3
2.1. Diagrama central del sistema. Sistema de monitoreo de eventos.	7
2.2. Zoom de nuestro sistema de monitoreo de eventos.	7
2.3. Rango de los valores de los píxeles de profundidad.	10
2.5. Tracking: el esquema del enfoque utilizado.	14
2.6. Zoom en el <i>tracker</i> de apariencia discriminatorio	15
2.7. <i>Feet detection</i>	18
2.8. Resultado al graficar la salida del algoritmo	18
3.1. Reconocimiento de eventos en nuestro SME.	19
3.2. Vista general de como se llega a la detección de eventos.	20
3.3. Zonas contextuales de la escena experimental	21
4.1. Diagrama central del sistema. Etapa de clasificación.	27
4.2. Esquema de evaluación y clasificación de los pacientes.	27
4.3. Paciente realizando tareas guiadas	28
4.4. Paciente realizando una de las tareas semi-guiadas	29
4.8. Técnica <i>k-fold cross-validation</i>	39
5.1. Diagrama central del sistema. Diagnóstico definitivo.	41
5.2. Sistema de Monitoreo de Eventos	42
5.4. Enfoque <i>wrapper</i> para la selección de un subconjunto de características.	44
5.5. Evaluación de la autonomía y clasificación del diagnóstico de demencia	45
5.7. Coeficiente de correlación en ambas tareas.	47

Capítulo 1

Introducción

La presente tesina de grado para la obtención del grado de Licenciado en Ciencias de la Computación del estudiante Álvaro Gómez Uría^{1,4} es un trabajo de investigación realizado en el centro de investigación INRIA (Instituto Nacional de Investigación en Informática y Automática)⁴, bajo las órdenes como director del Dr. François Brémont^{2,4} y de los estudiantes de postdoctorado Carlos Fernando Crispim-Junior^{2,4} y Alexandra König^{2,3}.

Gran parte de este trabajo fue presentado y aprobado en el *Journal Frontiers in Aging Neuroscience* (Alexandra König y col. 2015). Por estos motivos, se pueden encontrar algunas expresiones en inglés tanto en el texto como en las imágenes ya que se ha decidido conservarlas.

¹Universidad Nacional de Rosario, UNR. ²EA CoBTeK, Université Côte d'Azur (UCA), Nice, France. ³Alzheimer Center Limburg, Maastricht University Medical Center, School for Mental Health and Neuroscience, Maastricht, Netherlands. ⁴STARS, INRIA, Sophia Antipolis, France

1.1. El problema a abordar

Actualmente, la evaluación de la autonomía y de las habilidades funcionales involucra escalas de clasificación clínica. Sin embargo, las escalas están, con frecuencia, limitadas en su capacidad de proporcionar información sensible y objetiva. Por el contrario, las tecnologías de información y comunicación pueden superar esas limitaciones capturando completamente la funcionalidad, así como también las alteraciones cognitivas asociadas con la Enfermedad de Alzheimer¹ (EA).

En este trabajo se propone investigar una evaluación cuantitativa de la autonomía y de la demencia de los pacientes mediante, no sólo un análisis de la marcha de los mismos, sino que además analizando el comportamiento de los participantes en las actividades instrumentales de la vida diaria² (AIVD) detectadas automáticamente por un sistema

¹Alzheimer disease - AD

²Instrumental activities of daily living - IADL

al cual denominaremos “Sistema de monitoreo de eventos de video” (SME) basado en el trabajo previo de Crispim y col. 2013. Es en este sistema donde se aplican los conocimientos de visión por computadora³ (detección de personas, seguimiento de objetos por trayectoria y *Feet detection*.).

Hemos trabajado con tres grupos de participantes divididos de la siguiente manera: pacientes con controles sanos⁴, pacientes que sufren un deterioro cognitivo leve⁵ y pacientes con la Enfermedad de Alzheimer. Estos han tenido que llevar a cabo en un escenario normalizado tareas físicas (*Single and Dual Task*. Ver Sección 4.1) y varias AIVD tales como la preparación de la medicación o hacer una llamada de teléfono (Ver Sección 4.1), mientras todo es grabado y monitoreado por una cámara RGB-D. Luego, los datos del sensor de video son procesados por el sistema de monitoreo de eventos que extrae automáticamente los parámetros cinemáticos de la marcha de los participantes y reconoce las actividades que son llevadas a cabo. Estos parámetros, más adelante denominados características (del Inglés “Features”), se utilizan para la evaluación de los niveles de desempeño de los participantes, que aquí se hace referencia como la autonomía, es decir, que tan “bien” o “mal” el paciente realizó las actividades, si pudo finalizarlas con éxito o no, etc. Además, son utilizados para presentar un diagnóstico de demencia del paciente.

La evaluación de la autonomía y del diagnóstico de demencia se plantean como una tarea de clasificación utilizando métodos de inteligencia artificial (más precisamente de *Machine Learning*) que toman como entrada los parámetros extraídos por el sistema de monitoreo de eventos y a través de una selección automática de dichos parámetros, junto con un clasificador brindan un resultado, que en el caso de la autonomía estará entre las clases “Bien”, “Mediocre” y “Mal”, mientras que en el diagnóstico de demencia serán “Healthy”, “MCI”⁵ o “Alzheimer”.

Los resultados obtenidos nos sugieren que es posible evaluar cuantitativamente el desempeño de las actividades instrumentales de la vida diaria y dar un diagnóstico de la demencia, con el apoyo de un sistema de monitoreo de eventos (La Figura 1.1 resume lo mencionado hasta el momento). Además, basado en los datos extraídos de los distintos grupos, pueden ser clasificados con gran precisión. Esto significa, que el uso de estas tecnologías puede proporcionar a los médicos un diagnóstico con información relevante para mejorar la evaluación de la autonomía y la demencia en tiempo real disminuyendo la parcialidad del observador (sesgos de investigación).

1.2. Enfermedad de Alzheimer y otras demencias seniles

Una de las características clave de la Enfermedad de Alzheimer es el deterioro en el funcionamiento diario, así como la disfunción ejecutiva debido a los cambios patológicos globales en las áreas frontales y posteriores (Marshall y col. 2006).

Estudios recientes muestran que en los pacientes con demencia la pérdida de funcionamiento en actividades instrumentales de la vida diaria está fuertemente asociado con

³Del inglés *Computer vision*

⁴*healthy controls*

⁵*Mild cognitive impairment* - MCI

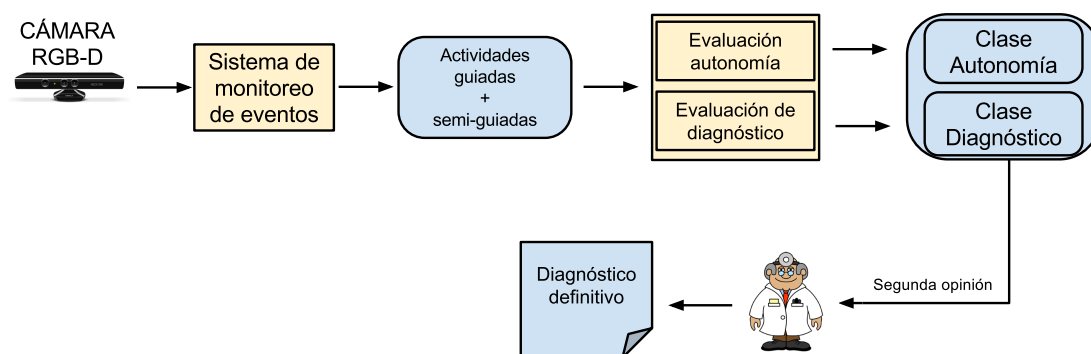


Figura 1.1: Diagrama central del sistema para la predicción de demencia.

un rápido deterioro cognitivo (Arrighi y col. 2005), y en particular, con pobres actuaciones en las tareas de función ejecutiva (Karzmark y col. 2012; Razani y col. 2007), como la Batería de Evaluación frontal (BAF) (Dubois, Slachevsky y col. 2000) o Prueba de la Toma de Rastro (versión B) (Tombaugh 2004). Por lo tanto, esto representa una predicción temprana para el deterioro cognitivo y posiblemente incluso para la conversión del deterioro cognitivo leve (MCI) a la Enfermedad de Alzheimer (EA) (Reppermund y col. 2013). Esto está en consonancia con los hallazgos más antiguos que muestran que la disminución de AIVDs están influenciadas por el funcionamiento cognitivo, y se ven afectados relativamente temprano en el transcurso de la demencia (Stern y col. 1990), y en particular, el componente ejecutivo en tareas AIVD que requiere la activación del lóbulo frontal superior (Baddeley y col. 1986).

Estado del arte sobre la clasificación y evaluación

La evaluación del funcionamiento en las actividades instrumentales de la vida diaria atrae gradualmente más atención en la investigación clínica y debe incluirse no sólo como parte de la evaluación de diagnóstico en demencias, sino que además sería esencial para evaluar la eficacia en contextos de rehabilitación (Clare y col. 2003; Cotelli, Calabria y Zanetti 2006).

La caracterización del deterioro en las AIVD es altamente discutible porque por el momento no existe ninguna norma en cuanto a la definición práctica o teórica (DeBettignies, Mahurin y Pirozzolo 1990). Por otra parte, hasta ahora, la evaluación de dichas actividades se limita principalmente a los cuestionarios y se basan a menudo en los reportes de los informantes o cuidadores, tal como la evaluación de la discapacidad para la escala de demencia (EDD) o la escala de las AIVD (Lawton y Brody 1969), que sufren de prejuicios e imprecisiones en las percepciones de los informantes, así como la posibilidad de que algunos adultos mayores no tengan a una persona que pueda comentar sobre el impacto del deterioro cognitivo en las actividades de su rutina. En general, las evaluaciones funcionales existentes carecen de la sensibilidad suficiente para detectar cambios funcionales sutiles o diferencias en el comportamiento, y por lo tanto, efectos en el trata-

miento (Gold 2012). Esto lleva a la necesidad urgente de una mejora en la medición de los cambios funcionales en las personas con alteraciones tempranas asociados a la enfermedad de Alzheimer en los estudios clínicos (Snyder y col. 2014). Además, sólo algunas de las herramientas nombradas capturan de manera temprana los déficits funcionales vistos en la EA.

El creciente reconocimiento de la necesidad de una medición más objetiva y directa ha llevado a algunos intentos de mejorar las evaluaciones de las AIVDs en la práctica clínica mediante el desarrollo de nuevos y extensos cuestionarios computarizados sobre AIVD basados en informantes (Sikkes, Lange-de Klerk, Pijnenburg, Gillissen y col. 2012) o la medición directa basada en el rendimiento (Moore y col. 2007), que difieren de los cuestionarios basados en informantes o de auto-informes tradicionales, como la escala de AIVD de Lawton (Lawton y Brody 1969), ya que observan directamente el hecho de un individuo llevando a cabo una AIVD, tal como hacer una llamada telefónica o administrar dinero.

Sin embargo, estos métodos pueden ser criticados, en primer lugar por ser fuertemente dependientes de un observador humano; en segundo lugar por la eliminación de la rutina elegida por el individuo y por las señales ambientales que normalmente facilitan la AIVD. Finalmente, la evaluación basada en el desempeño puede ser a menudo una pérdida de tiempo (Sikkes, Lange-de Klerk, Pijnenburg, Scheltens y col. 2009) y representa un único punto de datos de evaluación en comparación con las múltiples observaciones ofrecidas por un cuestionario que contiene comentarios sobre el comportamiento general de un individuo a través de las últimas semanas.

Las tecnologías de la información y comunicación, y en particular el análisis automático de videos de pacientes que llevan a cabo diversas AIVD podría ser un método de evaluación novedoso (Robert y col. 2013) para ayudar a superar esas limitaciones, reduciendo la variabilidad inter e intra-evaluador debido a la interpretación humana y aumentando el valor ecológico⁶ al eliminar completamente el observador humano del sitio de evaluación. Tales técnicas, y por lo tanto nuestra propuesta de evaluación automatizada de las AIVD basado en videos, difiere de estas herramientas actuales al permitir actuaciones y acciones de situaciones de la vida real de los pacientes capturados de forma remota en tiempo real, y que pueden ser evaluadas con precisión a fin de proporcionar al clínico una objetiva medición del desempeño y una “segunda opinión” sobre el estado general de la funcionalidad del paciente (tal como se indica en la Figura 1.1).

1.3. Propuesta

Los principales objetivos de este estudio son incursionar en el uso de la visión por computadora, en particular el análisis de video, y con el aprendizaje automático en el campo de la medicina para una evaluación cuantitativa de los pacientes en términos de autonomía y demencia, a través del análisis de patrones de comportamiento durante las actividades instrumentales de la vida diaria y la realización de algunas tareas físicas.

⁶Término clínico que en este caso se refiere a “cerca de la realidad”.

Para conseguirlo se intentará lograr un sistema confiable a la hora de detectar personas y las acciones que los mismos realizan, para, a partir de esa información, poder crear un modelo que abarque cada uno de los grupos de estudio y así llevar a cabo la predicción y posterior diagnóstico del paciente.

Visión por Computadora

La visión por computadora es una rama muy ligada a la inteligencia artificial. Desarrolla tanto la teoría como la tecnología necesaria para emular la percepción visual humana. Su principal objetivo es construir sistemas artificiales capaces de interpretar escenas naturales o datos multidimensionales (Meré 2008). Dentro de las técnicas que existen en la visión por computadora, vamos a utilizar la detección de personas, el seguimiento de objetos por trayectoria⁷ y el *Feet detection* (Ver Figura 2.2).

La detección de personas desarrollada se basa en la diferencia de *background* entre dos imágenes consecutivas (A.-T. Nghiem y Bremond 2014). La trayectoria sirve para determinar el seguimiento de la persona detectada y para el proceso de re-identificación (Chau y col. 2014). Por último, el *Feet detection* está basado en la medición de la velocidad de una persona mientras se encuentra en una cinta caminadora (Fosty, Sadoun y Bremond 2015). Dichas técnicas interpretan y evalúan la actuación de los pacientes.

Aprendizaje automático

El aprendizaje automático, más conocido como *Machine Learning*, es una rama de la inteligencia artificial cuyo principal objetivo es lograr que las computadoras sean capaces de realizar predicciones a partir de un modelo que represente las posibles soluciones. Los modelos y los métodos de entrenamientos generalmente se clasifican en supervisado y no-supervisado. En el aprendizaje supervisado, la base de conocimiento del sistema está formada por ejemplos etiquetados. De esta manera, el algoritmo produce una función que establece una correspondencia entre las entradas y las salidas deseadas del sistema. En cambio, el aprendizaje no-supervisado no contiene una base de conocimiento etiquetada, por lo tanto el sistema tiene que ser capaz de reconocer patrones para poder etiquetar dichas entradas. En nuestro trabajo utilizaremos los algoritmos de *Machine Learning* supervisados con el fin de clasificar las nuevas entradas, en este caso, nuevos pacientes.

Contribuciones

Tal como se mencionó en secciones anteriores, el sistema para la clasificación de la autonomía y la predicción de demencia consta de varios módulos y etapas, por lo tanto en esta sección se especificarán los contribuciones específicas realizadas por el alumno para finalizar el presente trabajo:

- Adaptación de las herramientas de detección de personas, seguimiento de objetos por trayectoria y de *Feet detection* para los videos específicos que conforman la base

⁷Comúnmente llamado tracking

de datos. Esto incluye: ciertas modificaciones en los códigos para que las detecciones y el seguimiento mantengan su eficiencia, y en algunos casos sean mejores.

- Modificación en el código de la ontología y su motor de inferencia desarrollados por el equipo de trabajo. Además del agregado de reglas específicas para la detección de las actividades que los pacientes realizan.
- Colaboración constante en la creación de la base de datos. Colocación de cámaras en el hospital en cuestión, más la creación de algunos demos para la explicación de su funcionamiento.
- Aprendizaje y aplicación de las técnicas de *Machine Learning* para la clasificación y evaluación de los pacientes. En este capítulo se centra el mayor esfuerzo y aporte de nuestro trabajo.

1.4. Organización del trabajo

El presente trabajo está organizado de la siguiente manera:

- En el capítulo 2, mostramos técnicas existentes de detección de personas, y en particular la utilizada en este trabajo. Luego hablamos sobre lo relativo al seguimiento de objetos por trayectoria; y por último presentamos la técnica de *Feet detection* utilizada en el sistema de monitoreo de eventos.
- En el capítulo 3, se presenta la ontología basada en restricciones usada para el reconocimiento de eventos de alto nivel en nuestro sistema.
- En el capítulo 4, tratamos las técnicas de aprendizaje automatizado (machine learning): los algoritmos de aprendizaje supervisado para la clasificación de los pacientes, la representación del comportamiento de los participantes y técnicas de selección de características.
- Y en el capítulo 5, finalmente nuestro caso de estudio en particular: las técnicas nombradas en los capítulos anteriores que forman nuestro sistema. Además, presentamos nuestra base de datos creada para el presente trabajo, el protocolo clínico a seguir por los pacientes y por último los resultados obtenidos.
- Las conclusiones y trabajo futuro se discuten en el capítulo 6.

Capítulo 2

Detección de Pacientes en Escena

En este capítulo vamos a desarrollar todas las herramientas necesarias para que nuestro sistema de monitoreo de eventos (SME) (Ver Figura 2.1) pueda detectar las actividades que los pacientes desarrollan y de esa manera poder dar una evaluación de la autonomía y un diagnóstico de demencia. Por lo tanto, presentaremos en detalle a todos aquellos módulos que conforman nuestro SME en particular: detección de personas, seguimiento de objetos por trayectoria (*tracking*) y *Feet detection* (Ver Figura 2.2).

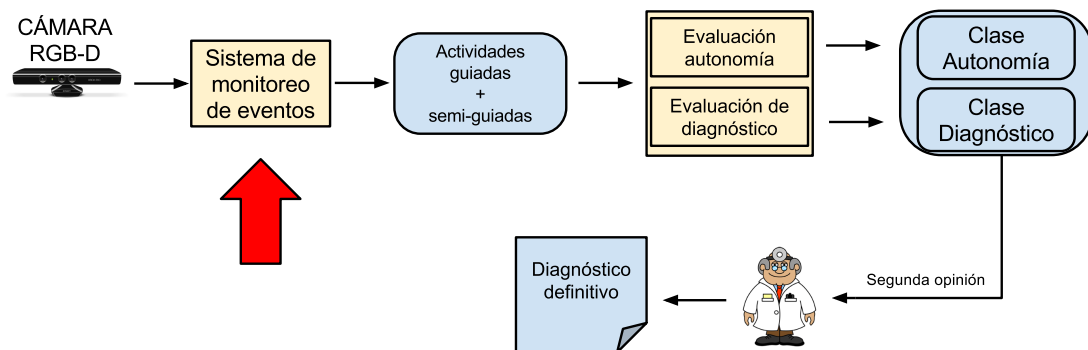


Figura 2.1: Diagrama central del sistema. Sistema de monitoreo de eventos.

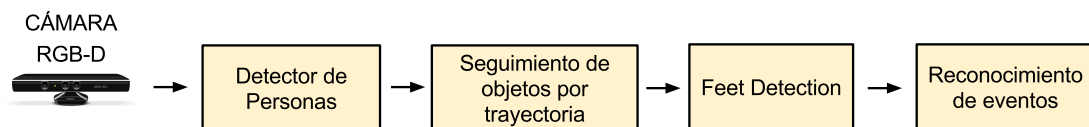


Figura 2.2: Zoom de nuestro sistema de monitoreo de eventos.

2.1. Detección de Personas en 3D

En el año 2010, Microsoft introdujo Kinect, la primera cámara RGB-D (*Red Green Blue + Depth*) de bajo costo en el mercado. Por cámara RGB-D se entiende a aquellas cámaras RGB con un nuevo sensor para la medición de la profundidad de los videos, el canal D. Estos dispositivos proveen esta información adicional como un conjunto de imágenes en escalas de grises dentro del rango $[0, 255]$ con una distribución lineal, donde a menor valor del píxel en cuestión implica mayor proximidad hacia la cámara. Junto con esta cámara, Microsoft también proporcionó una librería para la detección de personas y su estructura. Como se indica en Shotton y Fitzgibbon 2011, esta librería emplea un algoritmo de sustracción de fondo para detectar regiones en primer plano (*foreground*). Sin embargo, tal vez debido a una alta varianza en la medición de la profundidad cuando los objetos están demasiado lejos de la cámara, esta librería sólo detecta a las personas cuando están en el rango de 0,5 a un aproximado de 4,5 metros de las cámaras, es decir el rendimiento decae a partir de los 4.5 metros. Esto también fue mencionado en Spinello y Arras 2011. En nuestro caso de estudio se necesita que los pacientes sean detectados a una distancia mayor de los 4.5 metros mientras llevan a cabo distintos tipos de actividades dependiendo lo que el protocolo médico les indique. Por esta razón, este trabajo de investigación requirió el desarrollo de un nuevo *framework* de detección de personas que consista en un nuevo algoritmo de sustracción de fondo para manejar diferentes tipos de ruido en videos de cámaras RGB-D (A.-T. Nghiem y Bremond 2014), más un clasificador de personas, un seguimiento de objetos por trayectoria y un componente de eliminación del ruido. A este nuevo *framework* lo compararemos con los resultados que fueron obtenidos utilizando la librería de PrimeSense (la empresa detrás de las cámaras Kinect) para mostrar los avances en la detección de los pacientes en el rango entre 0.5 y 4.5 metros. Después de ese rango la librería deja de trabajar, mientras que nuestro *framework* continua detectando aunque con algunos ruidos.

Algoritmos de Substracción de Fondos

Para cámaras RGB, los algoritmos de substracción de fondos propuestos por Stauffer y Grimson 1999, Elgammal, Harwood y Davis 2000, Barnich y Droogenbroeck 2011, Hofmann, Tiefenbacher y Rigoll 2012 fueron utilizados con mucha frecuencia para la detección automática de regiones en primer plano (regiones en movimiento) en videos provenientes de cámaras fijas. Sin embargo, no ahondaremos en el tema ya que no serán utilizados en nuestro trabajo.

Para cámaras RGB-D, uno puede considerar a los videos de profundidad como un caso especial de videos en escalas de grises y aplicar los algoritmos de substracción de fondo de imágenes en escala de grises a los videos con profundidad. Sin embargo, debido a la alta varianza en las mediciones de la profundidad, el actual estado del arte de los algoritmos de substracción de fondo podría no tener buenos resultados en la clasificación de *foreground* / *background*.

Veamos como algunos de estos algoritmos de substracción de fondo pueden manejar la profundidad de los videos desde las cámaras RGB-D. En Stauffer y Grimson 1999 los

valores de cada píxel se modelan mediante una mezcla Gaussiana ponderada (*mixture of weighted Gaussian - GMM*). El valor de un píxel es clasificado como fondo si pertenece a una Gaussiana con un peso pesado; es decir se establece un umbral el cual determina si un píxel será clasificado como pesado o liviano y a partir de ello si es parte del *background* o *foreground*. Para aquellos videos de profundidad que contienen ruidos, es complicado de establecer el umbral que determina el peso Gaussiano para clasificar correctamente los píxeles. Además, si las personas se ubican cerca del *background*, solo una pequeña parte de sus cuerpos, relativamente distante del *background*, se detecta como *foreground*. Otro enfoque es modelar un *background* dinámico usando métodos no paramétricos. En El-gammal, Harwood y Davis 2000, los valores de cada píxel son modelados por la historia de los N valores más recientes. La construcción de modelos y la actualización se ha vuelto más simple, pero la selección de parámetros para distinguir *foreground* / *background* es difícil. En Barnich y Droogenbroeck 2011, Hofmann, Tiefenbacher y Rigoll 2012, los modelos de *background* también son un conjunto de N valores tomados de la entrada de los videos pero no de los valores más recientes. Estos N valores son tomados al azar de valores de píxeles entrantes. Cuanto más aparece el valor del píxel, hay mayor probabilidad de que se incluya en el modelo de *background*. Para los videos de profundidad con ruidos, parecido a lo que ocurre en Stauffer y Grimson 1999, estos algoritmos también sufren los mismos problemas, tales como la selección adecuada de los parámetros y distinguir a las personas cercanas al *background* de los valores ruidosos de profundidad del *background*.

```

1 process detectPersonUsingHOG: (window W, trainedModelOfPerson M){
2     (convolve(W, M) > UMBRAL): return YES ? return NO;
3 }

```

Código 2.1: Pseudocódigo del detector de personas

En resumen, para los videos con profundidad de cámaras RGB-D, la mayoría de los algoritmos de extracción de fondo actuales tratan de resolver el problema de la clasificación de *foreground* / *background* a nivel de píxel. En este nivel, la información es bastante limitada para tener una buena clasificación cuando la varianza de medición de la profundidad es demasiado alta.

Por lo tanto, se propone un algoritmo de sustracción de fondo específico para videos con profundidad de las cámaras RGB-D. El detector de personas no clasifica *foreground* / *background* a nivel píxel, pero provee suficiente información para remover el ruido. El ruido sólo es eliminado cuando se tiene toda la información de sustracción de fondo, la clasificación de personas basada en HOG, método de extracción de características basado en gradientes de histogramas (Dalal y Triggs 2005; A. Nghiem, Auvinet y Meunier 2012) (ver Código 2.1) y el seguimiento de objetos por trayectoria.

Algoritmo propuesto

Modelado de la alta varianza en la medición de la profundidad

La sustracción de fondo en este *framework* para la detección de personas, a diferencia de otros algoritmos de sustracción de fondo para cámaras RGB, no sólo asignan etiquetas de *foreground* o *background* a cada píxel, sino también etiquetas que corresponden a diferentes tipos de ruido específicos para videos con profundidad.

Para modelar la alta varianza en la medición de profundidad, se dividió el rango de varianza en tres regiones separadas: *background*, *flat noise*, y *flickering noise* como se muestra en la Figura 2.3.

Donde:

- la región de *background* corresponde a la región más densa de los valores de profundidad del *background*, es decir, d pertenece al *background* si $|d - d^0| \leq \tau_B$ donde d^0 es la media de los valores de profundidad en la región *background* y τ_B es un umbral determinado pequeño.
- *Flat noise* contiene a los valores de profundidad d que pertenecen al rango $\tau_B < |d - d^0| \leq \tau_F$ donde τ_F es 20 cm para nuestro caso.
- *Flickering noise* contiene a los restantes valores de profundidad de *background* que están muy desviados del valor medio de la región de *background*.

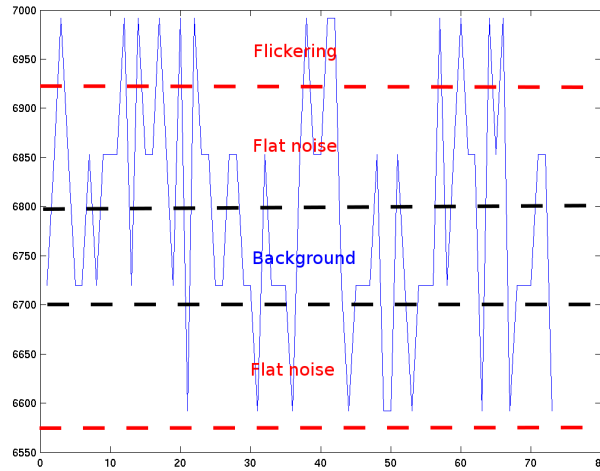


Figura 2.3: El rango de los valores de los píxeles de profundidad divididos en tres regiones: *background*, *flat noise*, y *flickering noise*

Modelo del *background*

El modelo del *background* dentro del algoritmo de sustracción de fondo se actualiza de forma selectiva con la ayuda del *feedback* proveniente del *framework*. Específicamente,

cuando el *framework* detecta a una persona, la región correspondiente a esa persona no es actualizada de modo que el algoritmo de sustracción de fondo pueda detectar a esa persona incluso cuando permanece en el mismo lugar durante mucho tiempo.

El propósito de nuestro modelo de *background* es reunir la información necesaria para poder modelar correctamente los tres tipos de regiones descritos anteriormente. Se modela los valores de profundidad en cada píxel por un conjunto de palabras de código (*codewords*). Cada *codeword* es una tupla (m, n, t_s, t_l, n_c) . En esta tupla, m es el valor medio de los valores de profundidad d que satisfacen la ecuación $|d - m^w| < \tau_B$ donde w es un *codeword* en particular. El valor n es la cantidad de valores de profundidad que satisfacen este *codeword*. t_s es el número de frame al momento de crear este *codeword*. En cambio, t_l es el número de frame de la última vez que el valor de profundidad satisfizo el *codeword*. Por último, n_c es la cantidad de veces que este *codeword* se cumplió de manera consecutiva.

La actualización de *codewords* es sencilla para la mayoría de las características de los *codewords*, excepto para el valor medio m . En nuestro algoritmo m se actualiza como sigue:

$$m_{new} = \frac{\min(N, n)m_{old} + d}{\min(N, n) + 1} \quad (2.1)$$

En nuestro caso, $N = 100$.

Clasificación de los valores de profundidad

La clasificación de *foreground* / *background* será en dos niveles: a nivel de píxel del algoritmo de sustracción de *background* y a nivel de objeto cuando el *framework* tenga la información necesaria para llevar a cabo la clasificación y el seguimiento por trayectoria.

A nivel píxel, el algoritmo de sustracción de fondo, dependiendo de los valores entrantes de profundidad, los etiqueta con: $\{BG, FL, FLK, TB, FG\}$. Las etiquetas *BG*, *FL* y *FLK* significan que los valores de profundidad están en las regiones graficadas en la figura 2.3 (*background*, *flat*, *flickering*). *TB* significa que los valores de profundidad pertenecen a un objeto desinteresado que recién aparece en la vista de la cámara (e.g., cuando se mueve una silla). *FG* corresponden a aquellos objetos a los cuales sus valores de profundidad no pertenecen a ninguna de las etiquetas anteriores. En otras palabras, estos valores de profundidad pertenecen a los objetos en movimiento.

Para explicar la clasificación, vamos a asumir que tenemos un conjunto de *codewords* $W = \{w\}$ para un píxel dado P y queremos clasificar un valor de profundidad d . En W , asumimos que w^0 es el *codeword* con el valor de profundidad más alto que lo satisface.

Para clasificar a d , el algoritmo primero busca el *codeword* w^* cuyo valor medio m^* esté cerca de d ($|d - m^*| < \tau_F$). Si no existe tal *codeword*, entonces se le asigna la etiqueta *FG*. Por otra parte, la etiqueta de d es determinada en base a las características de w^* . El algoritmo le asigna a d la etiqueta *BG* si w^* es también w^0 ; y le asigna la etiqueta *FL* a d si w^* no es w^0 pero $\tau_B < |m^* - m^0| \leq \tau_F$. La etiqueta *FLK* es asignada a d si w^* satisface dos condiciones: la primera es que cuando varios valores de profundidad consecutivos cumplen el mismo *codeword* w^* , la duración de esta consecutividad no

supere los β frames ($\beta n_c^* > n^*$, β es 3 en este caso). Y la segunda condición es que este fenómeno debe ocurrir al menos una vez cada γ cantidad de frames ($(t - t_s)/\gamma \leq n_c^*$, donde t es el tiempo actual. γ es 100 en este caso). Por último, el algoritmo le asigna la etiqueta *TB* a d si w^* satisface dos restricciones de *background* temporal como es el caso cuando se mueve una silla. La silla debe estar en la nueva posición un tiempo mayor a 200 ($n^* > 200$) para ser considerada como fondo. Y segundo, luego de que la silla fue movida al nuevo lugar, en cada píxel en la región correspondiente al nuevo lugar de la silla, el valor de profundidad medido por la cámara RGB-D es siempre el mismo, excepto cuando hay otro objeto que se esté moviendo ($(t_l^* - t_s^*)/n^* < \psi$ donde $\psi = 0,8$ en nuestro caso).

Eliminación del Ruido

Como se explicó anteriormente, a nivel píxel el algoritmo de sustracción de fondo no tiene suficiente información para remover el ruido. Por lo tanto, el ruido sólo es eliminado cuando hay información de la clasificación y el seguimiento. Podemos considerar dos tipos de ruido: las regiones del primer plano que contienen sólo los píxeles de ruido y las regiones del primer plano que contienen ambos, píxeles de ruido y a la gente real. Ambos tipos de ruido son eliminados en base a probabilidades de dimensiones posibles de personas, además dependen del clasificador de personas y tienen en cuenta el seguimiento de los mismos para poder diferenciar el ruido de un ser humano. Esto quiere decir que una persona al principio puede ser considerada como ruido pero en base a lo antes mencionado (su tamaño, seguimiento, etc) puede cambiar su clasificación; y viceversa.

Resultados del algoritmo propuesto

A continuación presentamos los resultados obtenidos por el algoritmo propuesto para la detección correcta de personas (ver Tabla 2.4). Para ello se presentan las definiciones de los términos *Precision* y *Recall*.

$$Precision = \frac{TP}{TP + FP} \quad (2.2)$$

$$Recall = \frac{TP}{TP + FN} \quad (2.3)$$

donde, TP: *True Positive rate*, FP: *False Positive rate*, y FN: *False Negative rate*.

	<i>PrimeSense</i>	<i>GMM</i>	Propuesto
<i>Precision</i>	0.979	0.979	0.998
<i>Recall</i>	0.987	0.987	0.995

Tabla 2.4: Desempeño del detector de personas en un rango menor a 4.5 metros.

2.2. Seguimiento de Objetos por Trayectoria (Tracking)

El seguimiento de objetos por trayectoria es una herramienta fundamental a la hora de clasificar a las personas como tales, y muy útil para la eliminación del ruido. Además, el seguimiento sirve para el proceso de re-identificación. Esto ayuda a que el objeto detectado se encuentre siempre bajo el mismo ID, y por ende se mantenga un correcto registro de su trayectoria. Para ello, nuestro sistema está basado en el trabajo propuesto por Chau y col. 2014 que desarrollaremos en las secciones siguientes.

No obstante, la selección de un algoritmo para el seguimiento de objetos no es una tarea fácil. Existen diferentes enfoques; veamos algunos: Kuo, C. Huang y Nevatia 2010 proponen un algoritmo basado en Adaboost para el aprendizaje de un modelo de apariencia discriminatorio para cada objeto móvil, con la complicación que el proceso Adaboost online consume mucho tiempo. En Santner y col. 2010 se presenta un *framework* que es capaz de seleccionar el *tracker* (algoritmo de seguimiento de objetos por trayectoria) más conveniente entre tres *trackers* predefinidos: *normalized cross-correlation (NCC)*, *mean-shift optical flow (FLOW)* y *online random forest (ORF)*. El enfoque es interesante pero la estimación online de la calidad del *tracker* no se menciona. También en J. H. Yoon, Kim y K. J. Yoon 2012 se propone un *framework* para el seguimiento de objetos por trayectoria integrando múltiples *trackers* basados en diferentes descriptores de características. Todos los *trackers* corren en paralelo. La salida de cada *tracker* es asociada con la probabilidad que representa su calidad. El *framework* selecciona el *tracker* con la mayor probabilidad. El hecho de correr varios *trackers* en paralelo es muy costoso en términos de tiempo de procesamiento, por lo que no son opciones adecuadas.

Por lo tanto, veamos las características del algoritmo utilizado.

Proceso del seguimiento de objetos

En la Figura 2.5 se presenta el esquema del proceso del seguimiento de objetos utilizado en este trabajo.

El enfoque toma a las imágenes de video y a la detección de objetos como entrada. Para cada frame, utilizando *KLT feature tracker* (Shi y Tomasi 1994) se estima si se detecta correctamente un objeto. Los objetos detectados correctamente tienen en general una apariencia confiable. Por lo tanto, se utiliza un *tracker* de apariencia para el seguimiento de esos objetos a partir de descriptores determinados (Proporción del aspecto 2D, area 2D, etcétera) (Ver Sección 2.2). Los objetos incorrectamente detectados son regulados utilizando *KLT feature tracking* (Ver Sección 2.2) de forma tal de detectar correctamente dichos objetos o clasificarlos como ruido. Para estos objetos, el seguimiento se realiza mediante un *tracker* seleccionado entre el KLT y *trackers* de apariencia discriminatorio (ver Código 2.2). Esta selección ayuda a asegurarse una confiable correspondencia de objetos. Actúa en base a una selección de *trackers* online dependiendo los resultados que los mismos obtengan. Por último, a medida que el proceso va cumpliendo etapas descritas en la Figura 2.5, si se llega al final de la misma y el objeto no pudo ser clasificado, entonces se considera como ruido y es eliminado.

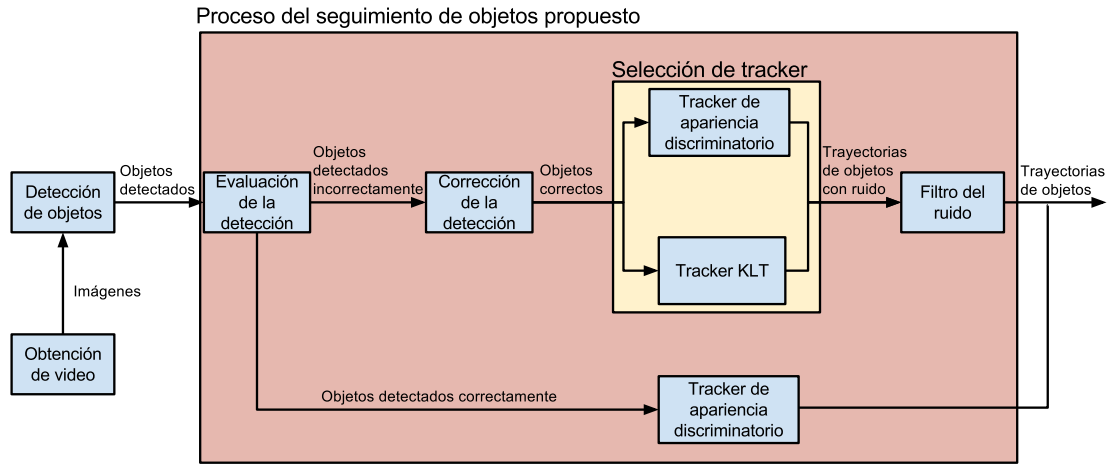


Figura 2.5: Tracking: el esquema del enfoque utilizado.

Evaluación de la detección

A través del uso de *KLT feature tracker* se le asignan etiquetas a los objetos detectados. Los mayores inconvenientes pueden surgir cuando se produce una oclusión o dos objetos están muy cerca entre ellos. De esta manera, si en el frame $t - 1$ el objeto tiene una etiqueta, y en el frame t el mismo objeto tiene más de una etiqueta, entonces se dice que está incorrectamente detectado. Caso contrario, la detección es correcta.

Corrección de la detección

Por cada etiqueta que se asigna al usar KLT feature, al objeto se le crea lo que en inglés se denomina “bounding box” (cuadro delimitador). Cuando la detección no es correcta, entonces el objeto tiene más de un bounding box, pero que en general son de menor tamaño que el objeto ya que por oclusión algunos puntos no han podido ser detectados. Por lo tanto, teniendo en cuenta el tamaño del objeto en frame anterior a la oclusión ($t - 1$), los bounding boxes son regulados, de manera tal que al final sólo quede uno.

Tracker de apariencia discriminadorio

Se propone un *tracker* de apariencia que se basa en la coherencia de cinco descriptores de apariencia del objeto:

- **Proporción del aspecto 2D:** la proporción entre el ancho y el alto del bounding box 2D del objeto.
- **Área 2D:** el area del bounding box 2D del objeto.
- **Histograma de color:** un histograma de color RGB normalizado de los píxeles en movimiento dentro de la caja objeto de delimitación.
- **Covarianza de color:** se utilizó el descriptor de covarianza propuesto en Bak y col. 2010. El descriptor de covarianza de color se computa combinando diferentes

características del objeto (ubicación, valores del canal RGB, magnitud del gradiente y orientación en cada canal).

- **Color dominante:** el color dominante se define en Yang y col. 2008. Es similar el descriptor del histograma de color con la diferencia que solo se tiene en cuenta los colores importantes del objeto.

Selección de tracker

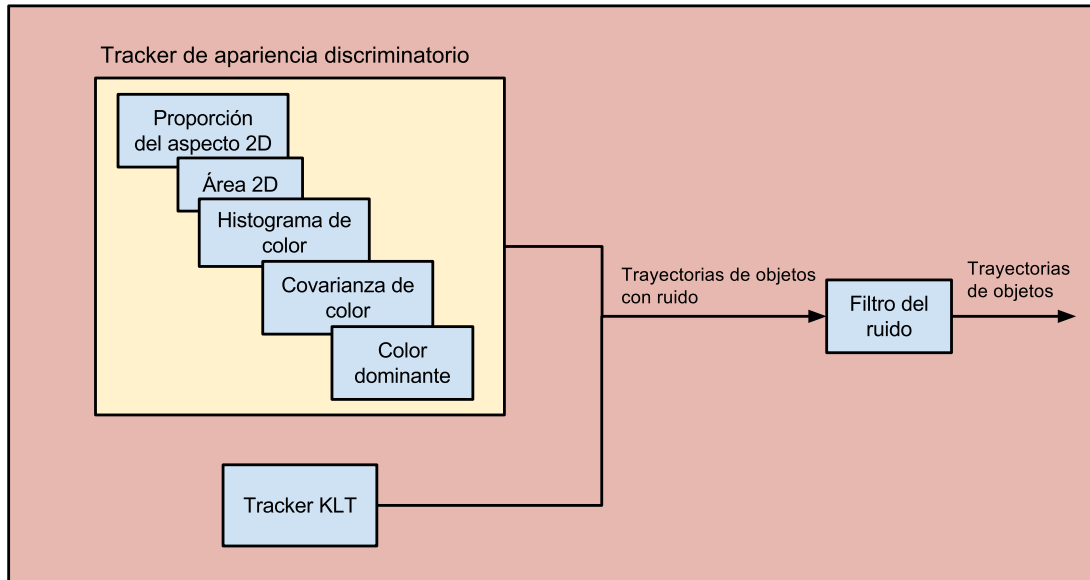


Figura 2.6: Zoom en el *tracker* de apariencia discriminador

Cada descriptor del objeto es efectivo en diferentes casos. Los descriptores de tamaño como la proporción del aspecto y el área son utilizados cuando los tamaños de los objetos móviles son distintos entre sí. Cuando los tamaños son similares, entonces los descriptor de color pueden ayudar a discriminar entre objetos. Cuando las condiciones de iluminación de la escena no son apropiadas, el descriptor de covarianza de color puede dar una mejor discriminación de objeto que los descriptores histograma de color y color dominante.

Tracker KLT

El segundo *tracker* para un objeto móvil se basa en el trabajo realizado por Shi y Tomasi 1994 que consta de las características de *Kanade-Lucas-Tomasi* (KLT). El *tracker* KLT toma a los objetos detectados como entrada e incluye tres pasos: La selección de características KLT, las características de seguimiento KLT y el seguimiento de objetos.

El objetivo de la selección de características KLT es detectar las características KLT situados en los objetos detectados utilizando los valores propios de sus matrices de gradiente. En la etapa de características de seguimiento KLT, cada feature KLT es seguido mediante la optimización de la traducción de sus características de ventana de punto. Por último, el objetivo del seguimiento de objetos es computar la trayectoria. Es en es-

te punto donde se aplica el algoritmo de Hungarian, método húngaro de optimización combinatoria que resuelve el problema de asignación en tiempo polinómico (Kuhn 1955).

```

1 process trayectoria: (Objetos 01, 02, ..., 0n-1){
2     for i=1; i<n-1; i++ {
3         Seguimiento 0i =
4             min_tracker(trackerAppearance(0i), trackerKLT(0i))
5     }
6
7     For each Seguimiento 0i apply Hungarian algorithm
8     return Trayectoria 0i
9 }
```

Código 2.2: Pseudocódigo del detector de personas

Filtración del ruido

Tal como se explicó en la sección 2.2, pueden aparecer más de un bounding box por objeto, lo cual se considera como ruido a aquellas que al final del proceso de seguimiento de objeto por trayectoria no pertenecían a un objeto deseado para el seguimiento, en nuestro caso, una persona. Estos objetos ruidosos son eliminados al final del proceso.

2.3. Feet Detection

Diferentes grupos de investigadores se centraron en la búsqueda de algunos patrones a través del análisis de la marcha en pacientes adultos mayores con la enfermedad de Alzheimer (AD) o un deterioro cognitivo leve (MCI). Tal fue el caso de Maquet y col. 2010, quienes han concluido que las personas que sufren AD y MCI presentan una alteración en algunas variables medidas durante las tareas físicas denominadas Single y Dual Task (Ver Sección 4.1). Por ejemplo, los pacientes con AD tienen una longitud menor en las zancadas y una disminución en la velocidad durante la realización del Dual Task. En Nadkarni y col. 2009 se concluye que los pacientes con los primeros signos de AD tienen una menor cadencia y pasos más cortos en comparación con pacientes sanos. Y por último, Muir y col. 2012 demostraron que durante la realización del Single Task no se presentan diferencias significantes entre los pacientes con AD, MCI y de control sano, sin embargo lo mismo no ocurre durante la realización del Dual Task, donde si se encontraron diferencias en los resultados.

Por lo mencionado anteriormente, sumado a que se evaluó el sistema sin un análisis de la marcha y los resultados no fueron alentadores, se decidió incorporar un *Feet detection* para poder enriquecer los perfiles de los participantes. El *Feet detection* agregado en nuestro sistema se basó en Fosty, Sadoun y Bremond 2015, el cual tuvo como objetivo la computación de la velocidad de una persona desempeñándose en una cinta motorizada para caminar. Se adaptó ese trabajo con el fin de detectar las piernas y las posiciones de los pies para finalmente tener un análisis del andar del paciente en las actividades físicas que ellos realizan.

Detección del suelo

Lo primero que se necesita para una buena detección de las piernas y de la posición de los pies de un participante, es un correcto funcionamiento en el detector de personas y el seguimiento por trayectoria del mismo. En nuestro caso, fueron descritos en las secciones 2.1 y 2.2 respectivamente. Luego, la eliminación de ruidos en las detecciones mejoran los resultados. Tal como el caso en donde algunos puntos pertenecientes al suelo pueden ser incluidos incorrectamente en la nube de puntos de la persona. En ese caso, el plano del suelo debe ser detectado para corregir la clasificación de los puntos. Basado en la parte más baja de la escena (con la asunción que los píxeles del suelo cubren al menos el 10 % de la imagen), la ecuación del plano es calculada como el plano que minimiza su distancia a los puntos del suelo (ver Código 2.3). Una vez que esta ecuación del plano ha sido estimada, la nube de puntos de la persona puede ser filtrada y removidos los puntos pertenecientes al suelo.

```

1 process groundDetection: (video V){
2     if (lowestPoint(V) >= 10 % of the image)
3         return GROUND_DETECTED
4     Otherwise
5         re-run groundDetection
6 }
```

Código 2.3: Pseudocódigo del detector del suelo

Detección de la piernas y de la posición de los pies

Luego del proceso de filtrado de ruido antes mencionado, sólo la parte más baja de la persona es conservada (el 25 % de los puntos más bajos). Cada punto que queda entonces es clasificado como perteneciente a la pierna derecha o a la izquierda, en función de la distancia horizontal entre los extremos derechos e izquierdos de la nube (ver Figura 2.7). Ambas piernas se dividen verticalmente en mitades de la misma altura y la línea recta que pasa por el centro de gravedad de cada parte (ver Figura 2.7 los puntos anaranjados y línea recta) representa la estructura de las piernas. Por último, los pies son las proyecciones de los puntos más bajos de las piernas en estas líneas rectas (ver Figura 2.7 los puntos amarillos).

Obtención de parámetros de la marcha

Una vez que el sistema detecta las piernas del paciente y por ende, la posición de los pies en sucesivos *frames*, al graficar la salida del algoritmo obtendremos algo similar a lo que aparece en la figura 2.8. Lo que nos muestra esta figura es la distancia entre pies, en este caso dada en milímetros pero fácilmente transformable a metros. A partir de esa información, hemos sido capaces de calcular los parámetros de la marcha de los pacientes para poder potenciar la información obtenida hasta el momento.

Tal como se muestra en la Figura 2.8, las cruces verdes son los mínimos locales, mientras que los puntos rojos son las máximos locales. Los mínimos locales se dan en

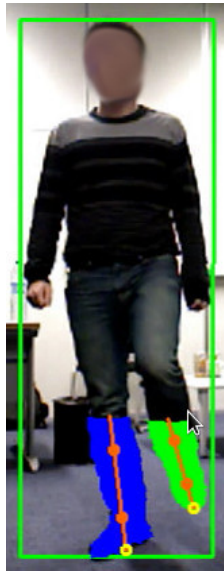


Figura 2.7: *Feet detection*

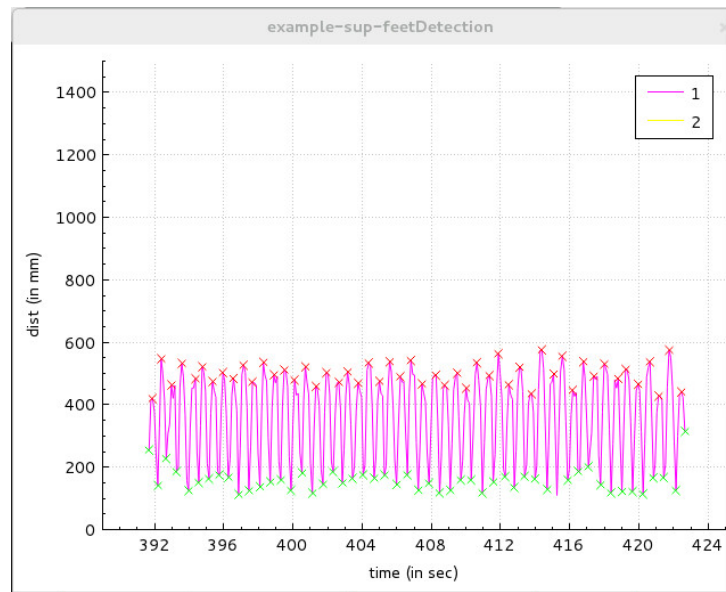


Figura 2.8: Resultado al graficar la salida del algoritmo

el momento en que al dar un paso, el pie que está en el fondo al trasladarse hacia adelante pasa cerca del otro pie. Mientras que los máximos locales son aquellos en donde la distancia entre los pies es la máxima; justamente al momento de finalizar un paso.

Por lo tanto, con los mínimos y máximos se pueden obtener la cantidad de pasos, las zancadas, sus dimensiones, la velocidad instantánea, cantidad de pasos realizados y todos aquellos parámetros que puedan ser derivados de los mencionados anteriormente (ver Código 2.4).

```

1 process gaitParameters: (feetDetection FD){
2     List local_max = maximos(FD);
3
4     numberOfSteps = local_max.size();
5     ...
6 }

```

Código 2.4: Pseudocódigo para la obtención de los parámetros de la marcha

Capítulo 3

Reconocimiento de Actividades

Para completar nuestro sistema de monitoreo de eventos para la detección temprana de la Enfermedad de Alzheimer (EA) necesitamos no sólo detectar una persona, sino que además poder reconocer los tipos de actividades que llevan a cabo. Para eso hemos necesitado agregar un módulo llamado Reconocimiento de Eventos encargado de ésta tarea. Por lo tanto, será el turno de desarrollarlo en este capítulo (Ver Figura 3.1).

Dichas actividades a realizar fueron previamente determinadas por un grupo de médicos especialistas en el tema (el protocolo clínico se encuentra en la sección 4.1).

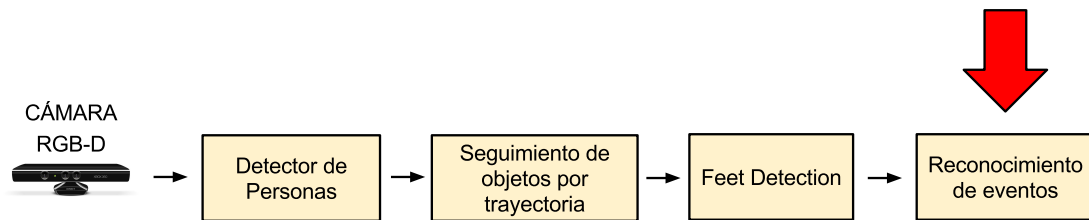


Figura 3.1: Reconocimiento de eventos en nuestro SME.

3.1. Eventos

Los eventos son modelados y reconocidos utilizando un lenguaje de ontologías declarativo basado en restricciones y el algoritmo de inferencia propuesto por T. Vu y Thonnat 2003. El lenguaje emplea una terminología natural para permitirle a los usuarios finales (por ejemplo, expertos médicos) agregar y cambiar fácilmente la base de conocimientos de eventos del sistema (tipo especial de base de datos para la gestión del conocimiento). Los modelos de eventos son creados teniendo en cuenta el conocimiento *a priori* sobre la escena experimental, los objetos físicos del mundo real reconocidos por el sistema de visión (por ejemplo, una persona) o definidos *a priori* (como ser el caso de una cama, una silla, lavabo, etc.) y por el conocimiento de dominio. Se utilizan modelos de objetos físicos para conectar la salida de los algoritmos visuales con los modelos de eventos.

La inferencia de eventos se realiza mediante un algoritmo temporal que sigue un enfoque bottom-up (de abajo hacia arriba). Toma como entrada el conjunto de personas detectadas y seguidas por los algoritmos de visión; los vincula como instancias de los modelos de objetos físicos adecuados y luego infiere instancias de eventos sobre la satisfacibilidad de las restricciones de los modelos de eventos definidos (ver Figura 3.2).

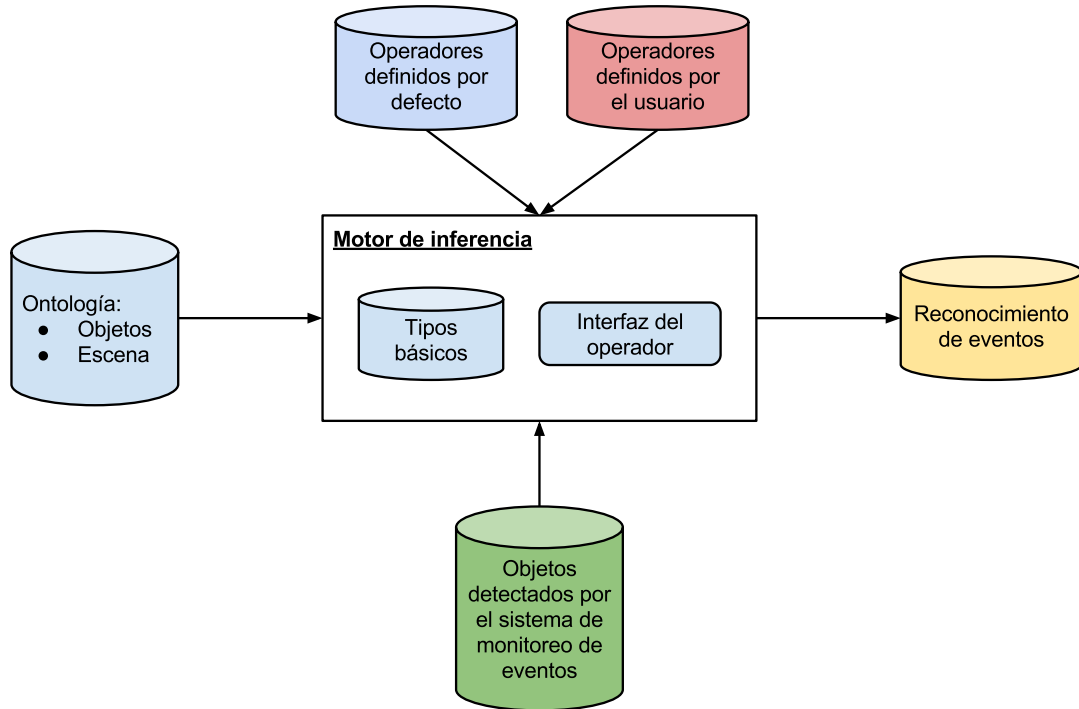


Figura 3.2: Vista general de como se llega a la detección de eventos.

3.2. Composición de un modelo de eventos

Un modelo de evento se compone por tres partes principales: los objetos físicos, los componentes (sub-eventos) y las restricciones. Los objetos físicos se refieren a los objetos del mundo real que participan en el evento modelado. El conjunto de tipos de objetos físicos depende del dominio para el cual la tarea de modelado de eventos es aplicada. Cinco tipos de objetos se definen para este enfoque:

1. Móvil
2. Persona
3. Zona contextual
4. Equipo contextual
5. La escena

Móvil es una clase genérica que contiene el conjunto básico de atributos de cualquier objeto en movimiento detectado en la escena (por ejemplo: la posición 3D, la anchura

y la altura, etc.). *Persona* es una extensión de la clase móvil cuyos atributos son la postura del cuerpo y la apariencia. El tipo de objeto *escena* permite al framework modelar eventos relacionados con las características de la escena monitoreada como un todo, por ejemplo la cantidad de personas en la escena. *Las zonas y equipo contextuales* se refieren al conocimiento *a priori* sobre la escena (por ejemplo, las zonas de la cocina y del sillón, o a los objetos televisor o pava) (ver Figura 3.3). Se componen de la descomposición de una proyección 3D de la escena en el plano del suelo en un conjunto de zonas espaciales que llevan la información semántica sobre la escena monitorizado (por ejemplo, zonas como “TV”, “sillón”, “escritorio”, “máquina de café”), o de la información derivada de algoritmos de visión, tales como el reconocimiento de objetos. En este trabajo adoptamos la primera opción ya que la mayor parte de la información semántica en la escena específica se refiere a los objetos sin movimiento (por ejemplo, muebles).

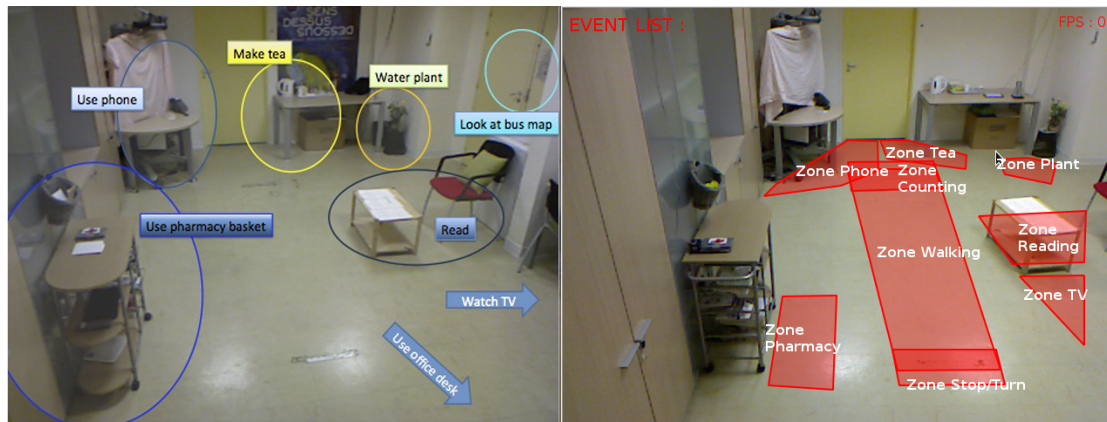


Figura 3.3: Zonas contextuales de la escena experimental

3.3. Restricciones

Las restricciones definen las condiciones que las propiedades físicas de objetos y/o componentes deben satisfacer. Pueden ser no-temporales, tales como las restricciones espaciales y de apariencia; o temporales, tales como el tiempo transcurrido entre dos sub-eventos (componentes). Las relaciones temporales están definidas utilizando el Álgebra de intervalos de James Allen, entre los que se destacan *BEFORE*, *MEET*, *AND* (Allen 1983).

La ontología categoriza jerárquicamente los modelos de eventos de acuerdo a su complejidad como (en orden ascendente): *primitive state*, *composite state*, *primitive event* y *composite event*. *Primitive state* modela el valor instantáneo de la propiedad de los objetos físicos (la postura de la persona, o si la persona está dentro de alguna zona semántica, entre otros). *Composite state* se refiere a la composición de dos o más *primitive states*. *Primitive event* modela un cambio en el valor de la propiedad del objeto físico (por ejemplo si la persona cambia de la postura de sentado a estar parado). Por último, *composite*

event se refiere a la composición de dos modelos de eventos previamente definidos que mantienen entre sí alguna relación temporal.

El reconocimiento de eventos sigue el algoritmo propuesto en T. Vu y Thonnat 2003, y trabaja de la siguiente manera:

- Vincula el concepto observable (por ejemplo, personas detectadas mediante un módulo de visión de bajo nivel) como una instancia de un objeto físico del lenguaje de la ontología (móvil, persona, etcétera).
- Las instancias de los objetos físicos luego son verificadas para chequear si sus atributos satisfacen las restricciones de los modelos de eventos que sólo dependen del frame en análisis (*primitive states*).
- Una vez que todos los *primitives states* son evaluados, las instancias recientemente inferidas siembran la verificación de las reglas de los modelos de eventos de alto nivel (*primitive* y *composite events*). La inferencia de eventos continua en un modelo *bottom-up* hasta que evalúa la satisfacción de las restricciones de todos los modelos de eventos que forman parte de la evidencia disponible.

3.4. Ejemplos de modelos

Modelo de un evento complejo

Para entender mejor la ontología y su funcionamiento descrito anteriormente vemos necesario la presentación de un modelo de un evento complejo que aunque no forma parte de nuestro sistema sirve como ejemplo didáctico.

```

1 CompositeEvent(BED_EXIT,
2     PhysicalObjects((p1:Person), (zB:Zone), (zSB:Zone))
3     Components(
4         (s1: PrimitiveState in_zone_bed(p1,zB))
5         (s2: PrimitiveState out_zone_bed(p1,zSB)))
6     Constraints((s1 meet s2)
7         (duration(s2) > 3))
8     Alarm ((Level : URGENT ))
9 )

```

Código 3.1: Modelo de BED_EXIT

El Código 3.1 describe como el sistema reconoce el evento *BED EXIT*¹. Este evento esta compuesto por tres objetos físicos (*PhysicalObjects*) (una persona y dos zonas semánticas), y dos componentes. El modelo componentes (*Components*) está formado por dos *primitive states*; la posición de la persona acostada en la cama, y estando fuera de la zona de la cama respectivamente (*in_zone_bed* y *out_zone_bed*). La persona *p1* corresponde a una persona dinámicamente detectada por los módulos de visión, y las

¹ *Bed exit* = Fuera de la cama

zonas zB y zSB corresponden a las zonas semánticas. En este caso en particular a las zonas de la cama y fuera de la misma, anotadas *a priori* en el sistema de coordenadas de la cámara 3D. La primera restricción (*Constraints*) determina que el sub-evento s1 tiene que estar seguido por el sub-evento s2 y sus intervalos temporales necesitan encontrarse (por eso usamos la relación temporal *MEET*), mientras que la segunda restricción establece una mínima duración de 3 segundos del sub-evento fuera de la zona de la cama. Los valores de los parámetros, como la duración mínima de la instancia de un modelo de un evento, se calculan basadas en las anotaciones de los eventos proporcionados por los expertos del dominio.

Modelos de nuestros eventos

Ya presentado el modelo de un evento complejo, ahora es el turno de dar a conocer algunos modelos de eventos de nuestro sistema. En particular vamos a mostrar todos los modelos que se necesitaron definir para poder capturar el evento *Single Task* (Tarea simple) que forma parte de las actividades guiadas del protocolo clínico (Sección 4.1).

```

1 PrimitiveState(Person_Inside_ZoneCounting,
2   PhysicalObjects((p1 : Person), (z1 : Zone))
3   Constraints ((p1->Position in z1->Vertices)
4     (z1->Name = zoneCounting))
5   Alarm ((Level : NOTURGENT))
6 )
7
8 PrimitiveState(Person_Inside_ZoneExerciseWalking,
9   PhysicalObjects((p1 : Person), (z1 : Zone))
10  Constraints ((p1->Position in z1->Vertices)
11    (z1->Name = zoneExerciseWalking))
12  Alarm ((Level : NOTURGENT))
13 )
14
15 PrimitiveState(Person_Inside_ZoneStop,
16   PhysicalObjects((p1 : Person), (z1 : Zone))
17   Constraints ((p1->Position in z1->Vertices)
18     (z1->Name = zoneStop))
19   Alarm ((Level : NOTURGENT))
20 )
21
22 CompositeEvent(changeZone_FromCountingToExerciseWalking,
23   PhysicalObjects((p1 : Person), (z1 : Zone), (z2 : Zone))
24   Components((c1 : PrimitiveState
25     Person_Inside_ZoneCounting(p1, z1))
26     (c2 : PrimitiveState
27       Person_Inside_ZoneExerciseWalking(p1, z2)))
28   Constraints((c1 meet c2)
29     (duration(c1) > 1)
30     (duration(c2) <= 1))
31   Alarm ((Level : NOTURGENT))
32 )

```

```

33
34 CompositeEvent(changeZone_FromExerciseWalkingToStop,
35     PhysicalObjects((p1 : Person), (z1 : Zone), (z2 : Zone))
36     Components((c1 : PrimitiveState
37         Person_Inside_ZoneExerciseWalking(p1,z1))
38         (c2 : PrimitiveState
39             Person_Inside_ZoneStop(p1, z2)))
40     Constraints((c1 meet c2)
41         (duration(c2) <= 2))
42     Alarm ((Level : NOTURGENT))
43 )
44
45 CompositeEvent(changeZone_FromStopToExerciseWalking,
46     PhysicalObjects((p1 : Person), (z1 : Zone), (z2 : Zone))
47     Components((c1 : PrimitiveState
48         Person_Inside_ZoneStop(p1,z1))
49         (c2 : PrimitiveState
50             Person_Inside_ZoneExerciseWalking(p1, z2)))
51     Constraints((c1 meet c2)
52         (duration(c2) <= 2))
53     Alarm ((Level : NOTURGENT))
54 )
55
56 CompositeEvent(changeZone_FromExerciseWalkingToCounting,
57     PhysicalObjects((p1 : Person), (z1 : Zone), (z2 : Zone))
58     Components((c1 : PrimitiveState
59         Person_Inside_ZoneExerciseWalking(p1,z1))
60         (c2 : PrimitiveState
61             Person_Inside_ZoneCounting(p1, z2)))
62     Constraints((c1 meet c2)
63         (duration(c2) <= 1))
64     Alarm ((Level : NOTURGENT))
65 )
66
67 CompositeEvent(WalkingTest_FirstAttempt_TOCONFIRM,
68     PhysicalObjects((p1 : Person), (z1 : Zone), (z2 : Zone))
69     Components((c1 : PrimitiveState
70         Person_Inside_ZoneCounting(p1, z1))
71         (c2 : CompositeEvent
72             changeZone_FromCountingToExerciseWalking(p1, z1, z2)))
73     Constraints((c1 meet c2))
74     Alarm ((Level : NOTURGENT))
75 )
76
77 CompositeEvent(WalkingTest_FirstAttempt,
78     PhysicalObjects((p1 : Person), (z1 : Zone), (z2 : Zone))
79     Components((c1 : CompositeEvent
80         WalkingTest_FirstAttempt_TOCONFIRM(p1, z1, z2))
81         (c2 : PrimitiveState
82             Person_Inside_ZoneExerciseWalking(p1,z2)))
83     Constraints((c1 meet c2))
84     Alarm ((Level : URGENT))
85     Miscellaneous((MergeStartEnd,

```



```

86         (Start WalkingTest_FirstAttempt_TOCONFIRM)
87         (Threshold 60)))
88 )
89
90 CompositeEvent(WalkingTest_FirstAttempt_COMPLETED ,
91     PhysicalObjects((p1 : Person), (z1 : Zone),
92                     (z2: Zone), (z3: Zone))
93     Components((c1 : CompositeEvent
94                 WalkingTest_FirstAttempt(p1, z1, z2))
95                (c2 : CompositeEvent
96                 changeZone_FromExerciseWalkingToStop(p1, z2, z3)))
97     Constraints((c1 meet c2))
98     Alarm ((Level : URGENT))
99     Miscellaneous((MergeStartEnd,
100                   (Start WalkingTest_FirstAttempt_TOCONFIRM)
101                   (Threshold 60)))
102 )
103
104 CompositeEvent(WalkingTest_SecondAttempt ,
105     PhysicalObjects((p1 : Person), (z1 : Zone), (z2: Zone))
106     Components((c1 : CompositeEvent
107                 changeZone_FromStopToExerciseWalking(p1, z1, z2))
108                (c2 : PrimitiveState
109                 Person_Inside_ZoneExerciseWalking(p1,z2)))
110     Constraints((c1 meet c2))
111     Alarm ((Level : URGENT))
112     Miscellaneous((MergeStartEnd,
113                   (End Person_Inside_ZoneStop)
114                   (Threshold 60)))
115 )
116
117 CompositeEvent(WalkingTest_SecondAttempt_COMPLETED ,
118     PhysicalObjects((p1 : Person), (z1 : Zone),
119                     (z2: Zone), (z3: Zone))
120     Components((c1 : CompositeEvent
121                 WalkingTest_SecondAttempt(p1, z1, z2))
122                (c2 : CompositeEvent
123                 changeZone_FromExerciseWalkingToCounting(p1, z2, z3)))
124     Constraints((c1 meet c2))
125     Alarm ((Level : URGENT))
126     Miscellaneous((MergeStartEnd,
127                   (End Person_Inside_ZoneStop)
128                   (Threshold 60)))
129 )
130
131 CompositeEvent(Single_Task ,
132     PhysicalObjects((p1 : Person), (z1 : Zone),
133                     (z2: Zone), (z3: Zone))
134     Components((c1 : CompositeEvent
135                 WalkingTest_FirstAttempt_COMPLETED(p1, z1, z2, z3))
136                (c2 : CompositeEvent
137                 WalkingTest_SecondAttempt_COMPLETED(p1, z3, z2, z1)))
138     Constraints((c1 before c2))

```

```
139 Alarm ((Level : URGENT))
140 Miscellaneous ((MergeStartEnd ,
141                 (Start WalkingTest_FirstAttempt_TOCONFIRM)
142                 (Threshold 60)))
143 )
```

Código 3.2: Todos los modelos para finalmente poder detectar el evento “Single_Task”

Si bien en el siguiente capítulo daremos detalles de las actividades que los pacientes tienen que realizar siguiendo el protocolo médico, para entender mejor el modelo descrito necesitamos contar en que se basa el evento *Single Task*. Se refiere a la primer parte de las actividades guiadas, donde el paciente camina ida y vuelta desde la zona de conteo hasta la zona de stop, pasando por la zona de caminata. A la ida la denominamos *First Attempt*, y a la vuelta *Second Attempt*. La palabra *Miscellaneous* que aparece en el código, es para captar las duraciones de los eventos. El evento *Dual Task*, el cual consiste en lo mismo que *Single Task* pero con el paciente contando hacia atrás desde el numero 305 será extraída de la misma manera pero con una excepción en el código que indique que *Dual Task* será siempre después de la tarea *Single Task*.

Capítulo 4

Clasificación de los pacientes mediante los eventos detectados

En este capítulo vamos a presentar la etapa de clasificación de los pacientes (ver Figura 4.1). Para ello es necesario presentar el protocolo médico que los pacientes tienen que realizar, la representación del comportamiento, la selección óptima de las características y los algoritmos de aprendizaje supervisado utilizados (ver Figura 4.2).

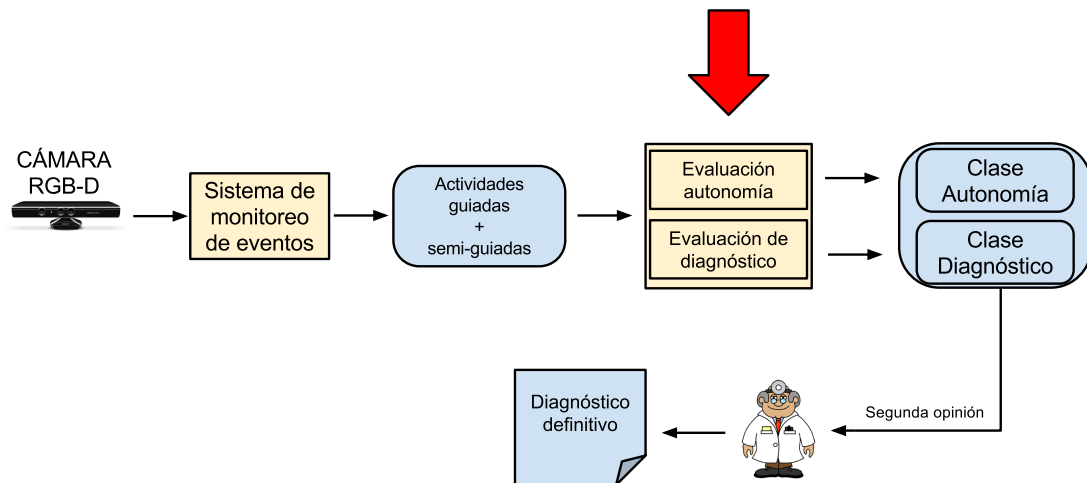


Figura 4.1: Diagrama central del sistema. Etapa de clasificación.

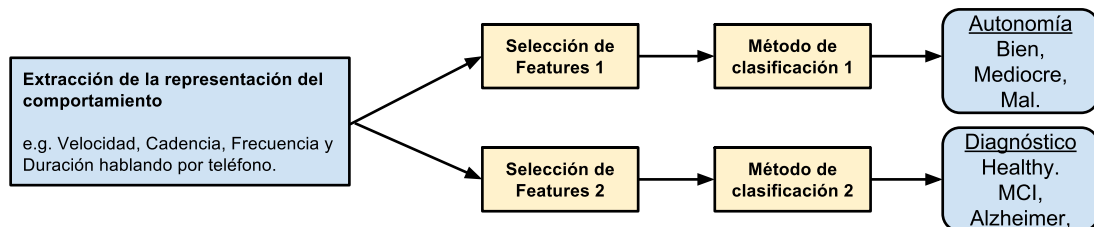


Figura 4.2: Esquema de evaluación y clasificación de los pacientes.

4.1. Protocolo médico

Para entender como se llevó a cabo la clasificación de los pacientes mediante los eventos detectados por nuestro sistema, es de vital importancia introducir el protocolo médico a realizar por los pacientes.

El protocolo médico se puede dividir en dos tipos de actividades: actividades guiadas y actividades semi-guiadas. En función al desempeño de los participantes en las actividades en términos de cantidad de actividades iniciadas y correctamente realizadas, así como repeticiones y omisiones, un médico definió la calidad de cada ejecución de la tarea. De acuerdo a esta verificación del rendimiento y basado en trabajos anteriores (Romdhane y col. 2012; Sacco y col. 2013; A. König y col. 2014), los participantes fueron agrupados (independientemente de su grupo de diagnóstico) en un desempeño *bueno*, *mediocre* y *malo* (la autonomía del paciente).

Actividades guiadas

Las actividades guiadas (ver fig. 4.3) (5 minutos) tienen la intención de evaluar los parámetros cinemáticos del análisis de la marcha de los participantes (por ejemplo, la duración, el número de pasos, la cadencia, longitud de zancada).

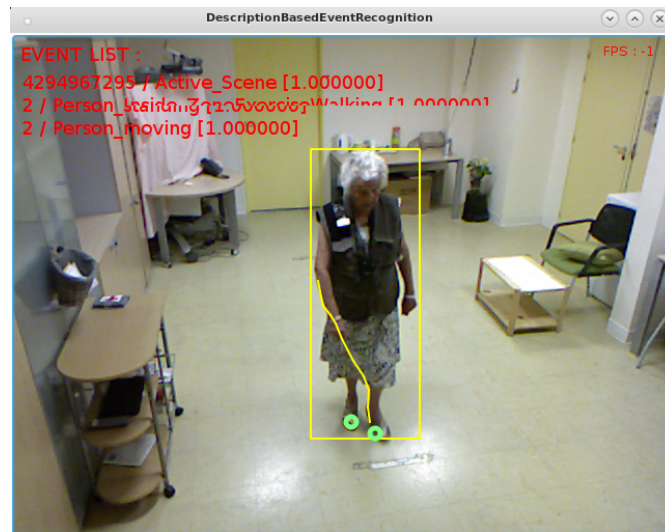


Figura 4.3: Paciente realizando tareas guiadas

Para ello los pacientes tienen que realizar dos pruebas. A la primer prueba la llamamos Single Task¹ y consiste en una caminata entre dos zonas previamente establecidas, llamadas zone counting² y zone stop (ver figura 3.3 para ver las zonas definidas en la escena). El paciente comienza en la zona de conteo (zone counting) y luego camina hasta

¹Single Task = Tarea única

²zone counting = zona de conteo

la zona de parada (zone stop). Una vez que alcanzó dicha zona, debe darse vuelta y volver a la zona de conteo. La segunda tarea es llamada Dual Task³. Tal como el nombre lo indica, es una tarea donde el paciente realiza dos funciones al mismo tiempo. Dichas tareas consisten en la tarea antes mencionada, Single Task, sumado a que durante su ejecución el paciente tiene que contar hacia atrás desde el número 305.

Actividades semi-guiadas

Las actividades semi-guiadas (ver fig. 4.4) (15 minutos) tienen como objetivo evaluar el nivel de autonomía del participante mediante la organización y la realización de una lista de actividades instrumentales de la vida diaria (AIVD). El participante debe ingresar solo a una sala de observación con la lista de actividades a llevar a cabo, y se le aconseja que tiene que abandonar la habitación cuando sienta que haya completado todas las tareas.

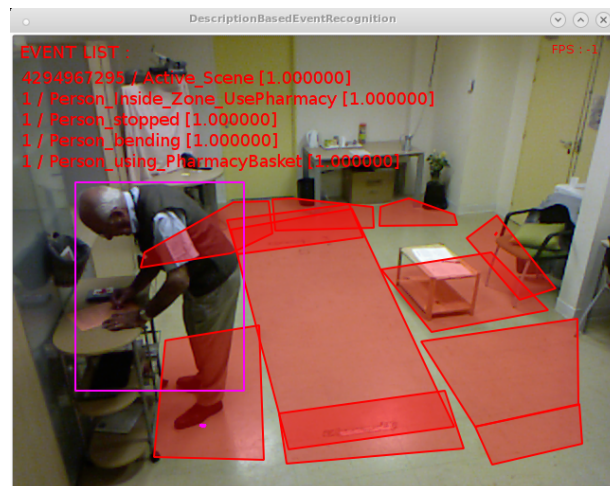


Figura 4.4: Paciente realizando una de las tareas semi-guiadas

La lista de las actividades instrumentales de la vida diaria son:

- Mirar televisión.
- Preparar te/café.
- Contestar el teléfono.
- Leer un periódico o una revista.
- Regar las plantas.
- Organizar los medicamentos prescritos dentro de una caja de medicamentos de acuerdo con el horario del producto diario / semanal.
- Llamar un taxi.
- Ver el recorrido de una línea de colectivo en un mapa.

³Dual Task = Tarea doble

4.2. Representación del comportamiento del participante

Ya presentadas las actividades del protocolo médico a seguir por los pacientes y nuestro sistema de monitoreo, a partir de este momento nos referiremos a la información extraída al final de las tareas como la representación del comportamiento. Se compone de distintos atributos en función al desempeño de las actividades. Más precisamente, dichos atributos pasarán a ser las características⁴ de nuestro sistema.

Por ejemplo, durante la realización de las actividades guiadas el sistema de monitoreo de eventos extraerá:

- Distancia recorrida
- Duración de la tarea
- Cantidad de pasos realizados
- El máximo, mínimo y el promedio de los pasos
- La varianza y la desviación estándar de los pasos
- Cadencia⁵
- Longitud de las zancadas⁶
- Cantidad de zancadas realizados
- El máximo, mínimo y el promedio de las zancadas
- La varianza y la desviación estándar de las zancadas
- Velocidad instantánea

Mientras que durante las actividades semi-guiadas se extraerá de cada actividad instrumental de la vida diaria:

- Duración en frames y en segundos
- Latencia
- Frecuencia

Donde la *frecuencia* se refiere a la cantidad de veces que un paciente realizó determinada actividad, y *latencia* de una actividad a la duración desde que el paciente inició la prueba, hasta que empezó dicha actividad.

Adición de características derivadas

Además, el uso de las características mencionados permitieron la adición de otros significativos para la clasificación de los pacientes. En otras palabras, hay ocasiones en las que nuevas características pueden ser derivadas de las ya existentes y presentar un mayor peso a la hora de la clasificación. Como por ejemplo de la frecuencia se puede determinar si el paciente se olvidó o no hacer alguna actividad específica.

También agregamos características, a los que denominamos parámetros Gaussianos de frecuencia y de duración de las actividades de la vida diaria en base al diagnóstico de demencia. Estos parámetros los obtuvimos de la siguiente manera (lo explicaremos solo con la frecuencia en la actividad “Regar Plantas”): obtenemos el valor medio y la desviación estándar de la frecuencia de “Regar Plantas” dividiendo a los pacientes por clases

⁴Característica: dato relevante para la resolución de una tarea computacional relacionada con una determinada aplicación.

⁵Cantidad de pasos por minuto

⁶Una zancada son dos pasos consecutivos

(*Alzheimer*, *MCI* y *Healthy*). Luego obtendremos tres nuevas características llamadas “Regar Plantas Probabilidad de frecuencia *Alzheimer*”, “Regar Plantas Probabilidad de frecuencia *MCI*” y “Regar Plantas Probabilidad de frecuencia *Healthy*”, los cuales son el resultado de la operación

$$Regar_PlantasProbFrecX = \frac{Frec - ValorMedioFrecX}{DesviacionEstandarFrecX} \quad (4.1)$$

donde X puede ser *Alzheimer*, *MCI* o *Healthy*

De esta forma, teniendo en cuenta las actividades instrumental de la vida diaria agregamos tres características en base a la frecuencia y otros tres en base a la duración por cada paciente.

Para finalizar hemos añadido información personal como el sexo y la edad de los pacientes. Una vez que hemos obtenido todas las características, se ejecuta un algoritmo de selección de características para mantener a aquellos más relevantes para una futura clasificación.

Lista completa de características

Age	Sex
Diagnosis	Performance
Single_Task Distance	Single_Task Total Duration
Single_Task durationForSpeedST	Single_Task Speed (Distance / Duration)
Single_Task (Total Duration - durationForSpeedST)	Single_Task Number of Steps
Single_Task Cadence	Single_Task Number of Strides
Single_Task Average Steps	Single_Task Max Steps
Single_Task Min Steps	Single_Task Variance Steps
Single_Task Standar Deviation Steps	Single_Task Average Stride
Single_Task Max Stride	Single_Task Min Stride
Single_Task Variance Strides	Single_Task Standar Deviation Strides
Double_Task Distance	Double_Task Total Duration
Double_Task durationForSpeedST	Double_Task Speed (Distance / Duration)
Double_Task (Total Duration - durationForSpeedST)	Double_Task Number of Steps
Double_Task Cadence	Double_Task Number of Strides
Double_Task Average Steps	Double_Task Max Steps
Double_Task Min Steps	Double_Task Variance Steps
Double_Task Standar Deviation Steps	Double_Task Average Stride
Double_Task Max Stride	Double_Task Min Stride
Double_Task Variance Strides	Double_Task Standar Deviation Strides
Centroid Single_Task Distance	Centroid Single_Task Duration
Centroid Single_Task durationForSpeedST	Centroid Single_Task Speed Average
Centroid Single_Task Speed (Distance / Duration)	Centroid Double_Task Distance
Centroid Double_Task Duration	Centroid Double_Task durationForSpeedST
Centroid Double_Task Speed Average	Centroid Double_Task Speed (Distance / Duration)
Ratio Duration Double and Single Task	Ratio Duration of only walk Double and Single Task
Ratio Speed Double and Single Task	Diff Duration Double and Single Task
Person_using_BusMap_Frequency of Event (times)	Person_using_BusMap_Expected
Person_using_BusMap_Freq_Prob_Poor	Person_using_BusMap_Freq_Prob_Good
Person_using_BusMap_Freq_Prob_Mediocre	Person_using_BusMap_Duration of Event (sec)
Person_using_BusMap_Dur_Prob_Poor	Person_using_BusMap_Dur_Prob_Good
Person_using_BusMap_Dur_Prob_Mediocre	Person_using_BusMap_Duration of Event (frames)
Person_using_BusMap_Latency (sec)	Person_using_PharmacyBasket_Frequency of Event (times)
Person_using_PharmacyBasket_Expected	Person_using_PharmacyBasket_Freq_Prob_Poor
Person_using_PharmacyBasket_Freq_Prob_Good	Person_using_PharmacyBasket_Freq_Prob_Mediocre
Person_using_PharmacyBasket_Duration of Event (sec)	Person_using_PharmacyBasket_Dur_Prob_Poor
Person_using_PharmacyBasket_Dur_Prob_Good	Person_using_PharmacyBasket_Dur_Prob_Mediocre
Person_using_PharmacyBasket_Duration of Event (frames)	Person_using_PharmacyBasket_Latency (sec)
Person_watering_Plant_Frequency of Event (times)	Person_watering_Plant_Expected
Person_watering_Plant_Freq_Prob_Poor	Person_watering_Plant_Freq_Prob_Good
Person_watering_Plant_Freq_Prob_Mediocre	Person_watering_Plant_Duration of Event (sec)
Person_watering_Plant_Dur_Prob_Poor	Person_watering_Plant_Dur_Prob_Good
Person_watering_Plant_Dur_Prob_Mediocre	Person_watering_Plant_Duration of Event (frames)
Person_watering_Plant_Latency (sec)	Person_reading_inChairReadingTable_Frequency of Event (times)
Person_reading_inChairReadingTable_Expected	Person_reading_inChairReadingTable_Freq_Prob_Poor
Person_reading_inChairReadingTable_Freq_Prob_Good	Person_reading_inChairReadingTable_Freq_Prob_Mediocre
Person_reading_inChairReadingTable_Duration of Event (sec)	Person_reading_inChairReadingTable_Dur_Prob_Poor
Person_reading_inChairReadingTable_Dur_Prob_Good	Person_reading_inChairReadingTable_Dur_Prob_Mediocre
Person_reading_inChairReadingTable_Duration of Event (frames)	Person_reading_inChairReadingTable_Latency (sec)
Person_making_Tea_Frequency of Event (times)	Person_making_Tea_Expected
Person_making_Tea_Freq_Prob_Poor	Person_making_Tea_Freq_Prob_Good
Person_making_Tea_Freq_Prob_Mediocre	Person_making_Tea_Duration of Event (sec)
Person_making_Tea_Dur_Prob_Poor	Person_making_Tea_Dur_Prob_Good
Person_making_Tea_Dur_Prob_Mediocre	Person_making_Tea_Duration of Event (frames)
Person_making_Tea_Latency (sec)	Person_using_Phone_Frequency of Event (times)
Person_using_Phone_Expected	Person_using_Phone_Freq_Prob_Poor
Person_using_Phone_Freq_Prob_Good	Person_using_Phone_Freq_Prob_Mediocre
Person_using_Phone_Duration of Event (sec)	Person_using_Phone_Dur_Prob_Poor
Person_using_Phone_Dur_Prob_Good	Person_using_Phone_Dur_Prob_Mediocre
Person_using_Phone_Duration of Event (frames)	Person_using_Phone_Latency (sec)
Number of event NOT done	Repeated events

Tabla 4.5: Lista de características

(Se presentan en inglés porque fue el idioma utilizado para realización del trabajo)

4.3. Selección de características

Muchos factores afectan el éxito del aprendizaje automático⁷ en una tarea determinada. En teoría, la idea de tener más características en los datos de entrenamiento⁸ indicaría algo más poderoso. Sin embargo, la experiencia práctica con los algoritmos de machine learning indican que éste no es siempre el caso. A fin de evitar el sobreajuste⁹ en los datos de entrenamiento, muchos algoritmos se basan en el principio de *Occam's Razor*, el cual sugiere que entre todas las hipótesis correctas, la hipótesis más simple es la que mejor capta la estructura del dominio del problema y tiene la más alta precisión en la predicción al clasificar nuevos casos, para construir un modelo simple que alcance un cierto nivel aceptable de rendimiento en los datos de entrenamiento (Gamberger y Lavrac 1997). Esta tendencia, a menudo lleva a que un algoritmo prefiera un pequeño número de atributos de predicción en lugar de un gran número de características que, si se usan en la combinación adecuada, son totalmente predictivas en la etiqueta de clase. Si hay demasiada información irrelevante presente o los datos son ruidosos y poco fiables, entonces el aprendizaje durante la fase de entrenamiento se torna más complicada. Por lo que llegamos a la técnica denominada en inglés *feature selection*¹⁰.

La selección de características es el proceso de identificar y eliminar la mayor cantidad de información irrelevante y redundante como sea posible. Esto reduce la dimensionalidad de los datos y puede permitir que los algoritmos de aprendizaje operen más rápido y con mayor eficacia. En algunos casos, la precisión en una futura clasificación puede mejorar, y en otros, el resultado es más compacto y fácil de interpretar la representación del concepto objetivo.

Existen principalmente dos enfoques a la hora de hacer la selección de características:

1. **Filter**: estos métodos utilizan heurísticas basadas en las características generales de los datos en lugar de un algoritmo de aprendizaje para evaluar el mérito de los subconjuntos de características.
2. **Wrapper**: las estrategias para la selección de características usan un algoritmo de inducción para estimar el mérito de subconjuntos de características. A menudo logra mejores resultados que *Filter* debido al hecho de que están sintonizados para la interacción específica entre un algoritmo de inducción y sus datos de entrenamiento. En su contra, los métodos de *wrapper* en la práctica pueden ser demasiado lentos para su uso práctico en los grandes dominios del mundo real que contienen muchas características. En cambio, los métodos de *Filter* son mucho más rápidos, ya que no implica la invocación repetida de un algoritmo de aprendizaje. Además, es posible la utilización del enfoque *wrapper* para diferentes clasificadores.

En nuestro trabajo nos vamos a centrar en la utilización de *wrappers* para clasificadores Bayesianos debido a que nos basamos en la hipótesis de que en el clasificador

⁷Aprendizaje automático = *Machine learning*

⁸Datos de entrenamiento = *Training data*

⁹Sobreajuste = *Overfitting*

¹⁰Selección de características

Naïve Bayes la distribución de probabilidad para los atributos son independientes entre sí (Kohavi y G. John 1997). Por lo tanto, el rendimiento en dominios con características redundantes se pueden mejorar mediante la eliminación de tales características. Este es el caso de nuestro sistema, ya que extraemos la mayor cantidad de características posibles y por eso podemos tener información redundante que produzca *overfitting* en los datos de entrenamiento (Mark A. Hall 1999) y por ende, que el modelo creado no sea el óptimo.

4.4. Algoritmos de Machine Learning

Existen diferentes tipos de algoritmos en el aprendizaje automático. Estos se agrupan en una taxonomía en función de la salida de los mismos. Los tipos de algoritmos más comunes son: los algoritmos de aprendizaje supervisado, no supervisado y semi-supervisados.

En este trabajo haremos hincapié en los algoritmos de aprendizaje supervisado, por lo tanto profundizaremos en estos algoritmos.

Aprendizaje supervisado

El aprendizaje supervisado se refiere al hecho de que al algoritmo se le provea un conjunto de datos a los cuales denominamos como “respuestas correctas”. Y a partir de estos, la tarea del algoritmo es la de generar más “respuestas correctas” a partir de nuevas entradas. Dentro de esta categoría de algoritmos dependiendo el tipo del valor del salida podemos diferenciar principalmente dos clases de problemas: de regresión, donde el resultado es un valor continuo, es decir un valor numérico; y de clasificación cuyo resultado es un valor discreto (cero ó uno), en otras palabras, una etiqueta de una clase. Nuestro trabajo pertenece a éste último tipo de problema ya que lo que buscamos es que a partir de nuevas entradas, en este caso pacientes, podamos determinar a que clase pertenecen (*Healthy*, *MCI* y *Alzheimer*, o *Bien*, *Mediocre* y *Mal*)

Veamos algunos ejemplos. El primero es un problema de regresión¹¹. Consta de la predicción del valor de las casas teniendo en cuenta la cantidad de metros cuadrado que tienen. Es decir, se introduce un conjunto de datos de entrenamiento de casas en los que para cada variable del conjunto de datos se dijo de antemano cuál era el precio correcto (el precio real al que se vendió esa casa) y por lo tanto, la tarea del algoritmo es la de generar el precio correcto para la casa nueva que se ingresó a la base de datos. Aquí, la salida del algoritmo es el precio, un valor continuo.

En cuanto a un ejemplo de un problema de clasificación¹¹ tenemos lo siguiente. Se quieren revisar expedientes médicos para tratar de predecir el cáncer de mama como benigno o maligno; en donde un tumor maligno es un tumor dañino y peligroso, y un tumor benigno es un tumor inofensivo. Tenemos un conjunto de datos recolectados y supongamos que tenemos el tamaño del tumor y si es benigno o maligno. Estos serían las respuestas correctas. Ahora, agregamos un nuevo dato con el tamaño del tumor y

¹¹Ejemplos extraídos del curso online que provee la Universidad de Stanford <https://www.coursera.org/course/ml>

la pregunta que necesitamos hacernos es: ¿Se puede determinar si el nuevo dato corresponde a un tumor maligno o benigno?. Por esto, éste es un ejemplo de un problema de clasificación. El término “clasificación” se refiere al hecho de que intentamos predecir un resultado de valor discreto: cero o uno, maligno o benigno.

Algunas veces en los problemas de clasificación se pueden tener más dos posibles valores como resultado. Por ejemplo, tal vez haya tres tipos de cáncer de mama, por lo que se intenta predecir el valor discreto cero, uno, dos o tres; en el que cero es benigno, es decir sin cáncer, y uno, dos y tres pueden ser tipos de cáncer en el caso de que sea maligno. Por lo tanto, esto también es un problema de clasificación, debido al conjunto de resultados del valor discreto que pertenecen al conjunto: sin cáncer o cáncer tipo uno, cáncer tipo dos o cáncer tipo 3.

En nuestro caso de estudio, tenemos las etiquetas de los pacientes de los datos de entrenamiento, esto quiere decir que por cada uno de ellos sabemos su autonomía, la cual la medimos en *bueno*, *mediocre* y *malo* y su diagnóstico de demencia entre *controles sanos*¹², *MCI*¹³ y *enfermedad de Alzheimer*¹⁴; esto nos ayudará para la predicción de la autonomía y del diagnóstico de demencia de las nuevos participantes con el fin de generar nuevas respuestas correctas.

4.5. Algoritmos de aprendizaje - Clasificadores

Un algoritmo de aprendizaje, o un algoritmo de inducción, forma descripciones de conceptos de datos de ejemplos. Descripciones de conceptos se refiere a menudo como al conocimiento o al modelo que el algoritmo de aprendizaje ha inducido a partir de los datos. El conocimiento puede ser representado de manera diferente entre un algoritmo y otro (Mark A. Hall 1999). Por ejemplo, C4.5 (Quinlan 1993) representa al conocimiento como un árbol de decisión; mientras que *Naïve Bayes* (Mitchell e Hill 1997) representa al conocimiento en forma de resúmenes probabilísticos.

Nuestro sistema está basado en el uso del algoritmo de clasificación *Naïve Bayes*, por lo que vamos a definirlo.

Clasificador *Naïve Bayes*

El algoritmo *Naïve Bayes* emplea una versión simplificada de la fórmula de Bayes para decidir a que clase pertenece una nueva instancia. La probabilidad posterior de cada clase se calcula teniendo en cuenta los valores de las características presentes en la instancia; la instancia se asigna a la clase con la más alta probabilidad. La ecuación 4.2 muestra la fórmula de *Naïve Bayes*, la cual hace la asunción de que los valores de las características son estadísticamente independientes dentro de cada clase (G. H. John y Pat Langley 1995).

¹²Healthy controls - HC

¹³MCI - Mild Cognitive Impairment. En español, Deterioro cognitivo leve

¹⁴Alzheimer Disease - AD

$$p(C_i|v_1, v_2, \dots, v_n) = \frac{p(C_i) \prod_{j=1}^n p(v_j|C_i)}{p(v_1, v_2, \dots, v_n)} \quad (4.2)$$

El lado izquierdo de la ecuación 4.2 es la probabilidad posterior de la clase C_i dados los valores de las características $\langle v_1, v_2, \dots, v_n \rangle$ observados en la instancia a ser clasificados. El denominador del lado derecho de la ecuación es con frecuencia ignorado ya que es una constante cuyo valor es fácil de computar si uno requiere que las probabilidades a posteriori de las clases sumen uno.

El aprendizaje con el clasificador de *Naïve Bayes* es sencillo e implica simplemente la estimación de las probabilidades en el lado derecho de la ecuación 4.2 de las instancias de entrenamiento. El resultado es un resumen probabilístico para cada una de las clases posibles. Si hay características numéricas, es una práctica común asumir una distribución normal - nuevamente los parámetros necesarios se estiman a partir de los datos de entrenamiento.

Una ventaja del clasificador *Naïve Bayes* es que solo requiere una pequeña cantidad de datos de entrenamiento para estimar los parámetros (las medias y las varianzas de las variables) necesarios para la clasificación.

Instancias #	Características				Clase
	Clima	Temperatura	Humedad	Viento	
1	soleado	caluroso	alto	falso	No jugar
2	soleado	caluroso	alto	verdadero	No jugar
3	nublado	caluroso	alto	falso	Jugar
4	lluvioso	templado	alto	falso	Jugar
5	lluvioso	frío	normal	falso	Jugar
6	lluvioso	frío	normal	verdadero	No jugar
7	nublado	frío	normal	verdadero	Jugar
8	soleado	templado	alto	falso	No jugar
9	soleado	frío	normal	falso	Jugar
10	lluvioso	templado	normal	falso	Jugar
11	soleado	templado	normal	verdadero	Jugar
12	nublado	templado	alto	verdadero	Jugar
13	nublado	caluroso	normal	falso	Jugar
14	lluvioso	caluroso	alto	verdadero	No jugar

Tabla 4.6: Base de datos para jugar al Golf

Vamos a presentar un ejemplo que nos ayude a explicar lo mencionado (Quinlan 1986). La tabla 4.6 muestra 14 instancias de días adecuados y no adecuados para que una persona pueda jugar Golf.

De la tabla 4.7 (a) a la tabla 4.7 (d) tenemos tablas de contingencia (posibilidad de que una cosa suceda o no suceda) mostrando la distribución de la frecuencia de las

	Jugar	No Jugar	
Soleado	2	3	5
Nublado	4	0	4
Lluvioso	3	2	5
	9	5	14

(a) Clima

	Jugar	No Jugar	
Alto	3	4	7
Normal	6	1	7
	9	5	14

(c) Humedad

	Jugar	No Jugar	
Caluroso	2	2	4
Templado	4	2	6
Frío	3	1	4
	9	5	14

(b) Temperatura

	Jugar	No Jugar	
Falso	3	3	6
Verdadero	6	2	8
	9	5	14

(d) Viento

Tabla 4.7: Tabla de contingencia compilada de la base de datos de Golf

relaciones entre las características y las clases de la base de datos 4.6. A partir de estas tablas es fácil calcular las probabilidades necesarias para aplicar la ecuación 4.2.

Ahora bien, imagine despertar una mañana y quiere determinar si el día es adecuado para jugar una partida de Golf o no lo es. Nota que el clima es soleado, la temperatura es calurosa, la humedad es normal y no hay viento (viento = falso). Por lo tanto, aplicamos la ecuación 4.2 y calculamos la probabilidad a posteriori de cada clase utilizando las probabilidades derivadas de las tablas 4.7 (a) hasta la (d):

$$\begin{aligned}
 p(\text{NoJugar}|\text{soleado}, \text{caluroso}, \text{normal}, \text{false}) &= p(\text{Nojugar}) \times \\
 &\quad p(\text{soleado}|\text{Nojugar}) \times \\
 &\quad p(\text{caluroso}|\text{Nojugar}) \times \\
 &\quad p(\text{normal}|\text{Nojugar}) \times \\
 &\quad p(\text{falso}|\text{Nojugar}) \\
 &= 5/14 \times 3/5 \times 2/5 \times 1/5 \times 2/5 \\
 &= 0,0069
 \end{aligned}$$

$$\begin{aligned}
 p(\text{Jugar}|\text{soleado}, \text{caluroso}, \text{normal}, \text{false}) &= p(\text{Jugar}) \times \\
 &\quad p(\text{soleado}|\text{Jugar}) \times \\
 &\quad p(\text{caluroso}|\text{Jugar}) \times \\
 &\quad p(\text{normal}|\text{Jugar}) \times \\
 &\quad p(\text{falso}|\text{Jugar}) \\
 &= 9/14 \times 2/9 \times 2/9 \times 6/9 \times 6/9 \\
 &= 0,0141
 \end{aligned}$$

Conclusión: con el día que nos levantamos SI deberíamos jugar al Golf ($0,0141 > 0,0069$).

Debido a la asunción de que los valores de las características son independientes dentro de las clases, la performance predictiva del clasificador *Naïve Bayes* puede verse adversamente afectada por la presencia de atributos redundantes en los datos de entrenamiento. Por ejemplo, si hay un feature X que es perfectamente correlativo con un segundo feature Y , entonces tratarlos como independientes significa que X (ó Y) tiene doble preponderancia en la ecuación 4.2 de lo que debería tener. Los investigadores P. Langley y Sage 1994 encontraron que la performance de *Naïve Bayes* mejora cuando son removidos las características redundantes. Sin embargo, Domingos y Pazzani 1996 fundamentan que, mientras que las correlaciones fuertes entre las características degradarán la performance, *Naïve Bayes* todavía puede funcionar bien cuando existen dependencias moderadas en los datos. La explicación para esto es que las dependencias moderadas se traducirán en una estimación imprecisa de la probabilidad, pero las probabilidades no están tan “mal” como para dar lugar a un aumento en los errores de clasificación.

4.6. Evaluación de los algoritmos

La evaluación del desempeño de los algoritmos de aprendizaje es un aspecto fundamental en el aprendizaje automático. No solo es importante con el fin de comparar los algoritmos competentes, sino que además es una parte integral del algoritmo de aprendizaje en sí. Una estimación de la precisión de la clasificación de nuevas instancias es el criterio de evaluación del desempeño más común, aunque otros basados en la teoría de la información han sido sugeridos (Kononenko y Bratko. 1993; Cleary, Legg y Witten 1996).

En este trabajo, la precisión de la clasificación es el criterio de evaluación primario para los experimentos usando previamente la selección de características con algoritmos de machine learning. La selección de características es considerada exitosa si disminuyendo la dimensionalidad de los datos al menos se mantiene o se mejora la precisión de los algoritmos de aprendizaje. La precisión de la clasificación se define como el porcentaje de ejemplos de prueba correctamente clasificados por los algoritmos. El porcentaje de error (una medida más comúnmente utilizada en estadísticas) de un algoritmo es $1 - \textit{precision}$. La medición de la precisión en un conjunto de prueba de ejemplos es mejor que usar datos de entrenamiento ya que los ejemplos en el conjunto de prueba no han sido usados para inducir las descripciones de conceptos. Utilizar el conjunto de datos de entrenamiento para determinar la precisión no es lo ideal ya que justamente el algoritmo está entrenado con esos datos.

Estrictamente hablando, la definición de precisión dada anteriormente es la precisión de muestra de un algoritmo. La precisión de muestra es una estimación de la (inmensurable) verdadera precisión del algoritmo. Esto es la probabilidad que el algoritmo clasifique correctamente una instancia extraída de una distribución desconocida X de ejemplos. Cuando no se cuenta con una gran cantidad de datos, es una práctica común el remuestreo de esos datos, es decir, particionar los datos en entrenamiento y prueba de diferentes maneras. Un algoritmo de aprendizaje es entrenado y testeado por cada partición y la

precisión final es el promedio. Hacer esto proporciona una estimación más fiable de la verdadera precisión de un algoritmo.

Random subsampling y *k-fold cross-validation*¹⁵ son los dos métodos más comunes de remuestreo (más conocido como *resampling*) (Geisser 1975; Schaffer 1993). Nosotros utilizamos *k-fold cross-validation* (ver fig. 4.8), el cual consta de dividir los datos de manera aleatoria en k subconjuntos mutuamente excluyentes de tamaño en lo posible iguales. Un algoritmo de aprendizaje es entrenado y testeado k veces; en cada iteración es testeado por una de las k partes y el resto de las partes ($k - 1$ folds) son usados para entrenamiento. En el *cross-validation* la estimación de precisión es el número total de clasificaciones correctas, dividido por la cantidad de ejemplos en los datos.

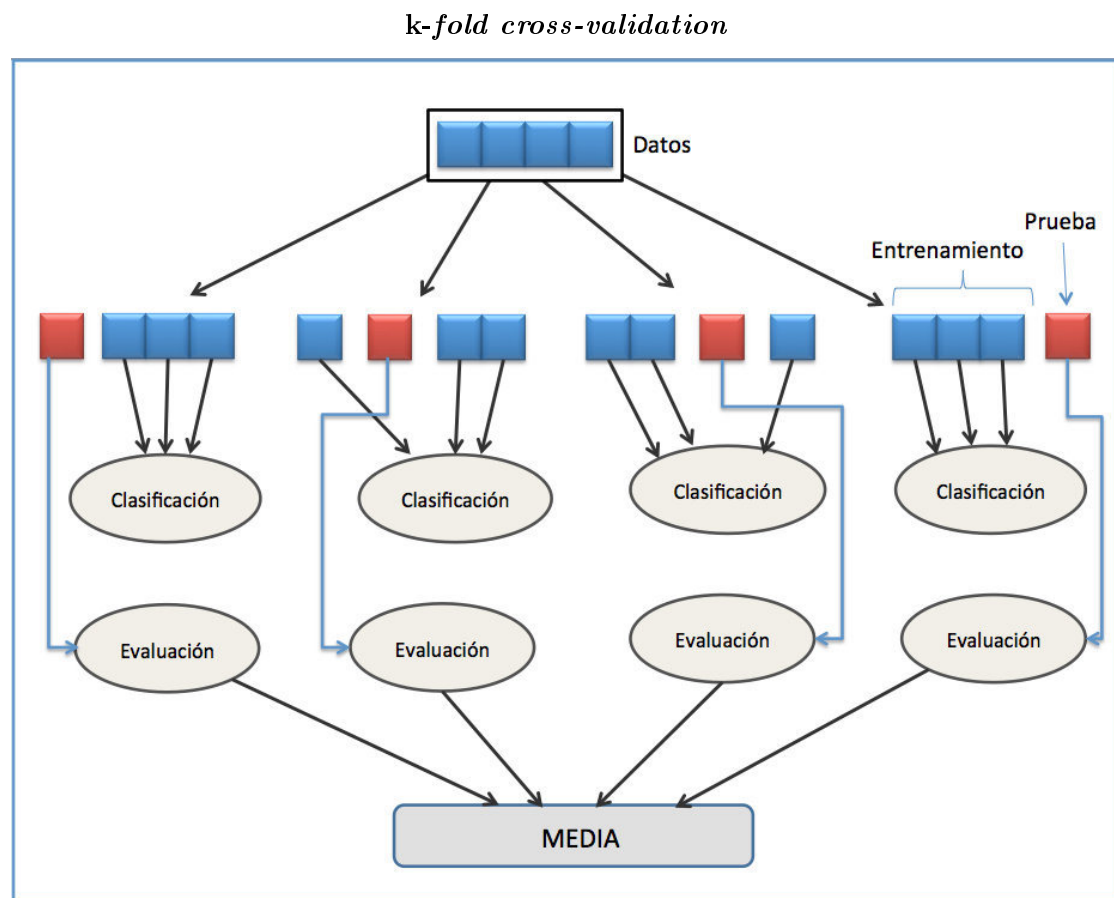


Figura 4.8: Técnica *k-fold cross-validation*. Previamente publicada en https://es.wikipedia.org/wiki/Validaci3n_cruzada

Dentro del método *cross-validation* existe un proceso llamado “Estratificación”. Esto significa que a la hora de dividir los datos en partes, la estratificación se asegura de

¹⁵En español, validación cruzada de k iteraciones

que la distribución en esa división sea pareja con respecto a las clases que la forman, es decir, se trata de que haya la misma cantidad de datos con características similares en las distintas partes. Ya que la estratificación ha demostrado que ayuda a reducir la varianza de la precisión estimada, especialmente para conjuntos de datos con muchas clases (Kohavi [1995](#)), lo hemos usado en nuestro trabajo.

Capítulo 5

Sistema para la predicción de demencia

En los capítulos anteriores se desarrolló todo lo necesario para que finalmente podamos obtener un sistema con el fin de que sirva como una herramienta de segundas opiniones para los médicos a la hora de la detección temprana de la enfermedad del Alzheimer o de un deterioro cognitivo leve en adultos, junto con una evaluación de su autonomía, mediante el desempeño en actividades diarias y tareas físicas, con el uso de una cámara RGB-D tal como se muestra en la Figura 5.1.

A continuación definiremos los últimos detalles de nuestro sistema. El mismo consta de un sistema de monitoreo de eventos, junto con la base de datos que construimos y como se llevó a cabo la evaluación de la autonomía y la demencia de los pacientes. Luego veremos la evaluación del sistema de monitoreo de eventos para mostrar confiabilidad del mismo y por último mostraremos los resultados obtenidos.

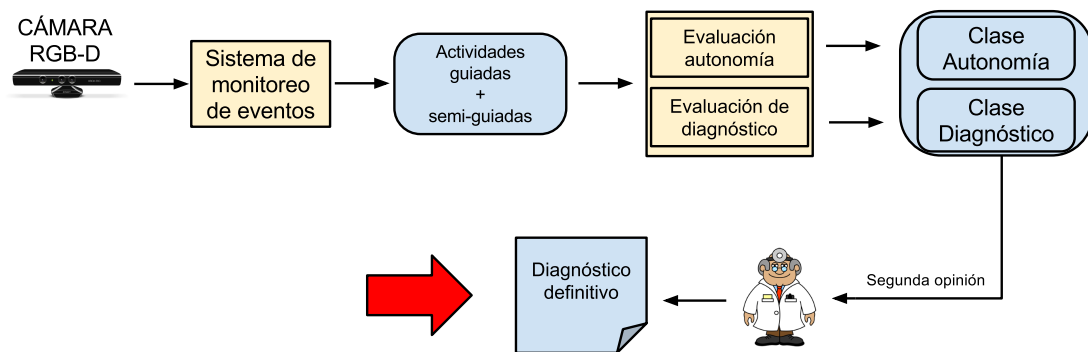


Figura 5.1: Diagrama central del sistema. Diagnóstico definitivo.

5.1. Sistema de Monitoreo de Eventos en módulos

El sistema de monitoreo de eventos (SME) puede ser descompuesto en los siguientes módulos (ver Figura 5.2): el detector de personas (Sección 2.1), el seguimiento de objetos por trayectoria (Sección 2.2), *Feet detection* (Sección 2.3) y el reconocimiento de actividades (Sección 3).

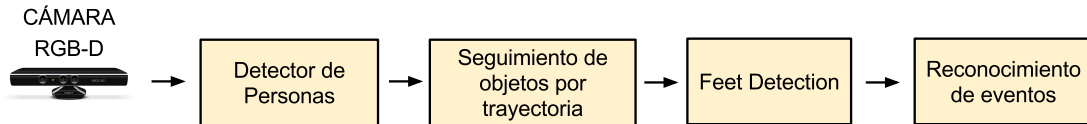


Figura 5.2: Sistema de Monitoreo de Eventos

La Figura 5.2 muestra al SME con una arquitectura en pipeline. En este caso, cada salida de un módulo es la entrada del siguiente módulo. Por lo tanto, cada uno depende del anterior y si bien funcionan los módulos por separado, solamente juntos tienen el potencial suficiente para lograr el objetivo propuesto. Cabe destacar que todo inicia con la información extraída por una cámara RGB-D.

5.2. Acerca los pacientes

Una parte más que importante del trabajo fue la de crear nuestra propia base de datos, para ello fueron reclutados participantes de más de 65 años dentro del protocolo Dem@Care en el Centro de Investigación de la Memoria de la ciudad de Niza, Francia ubicado en el departamento de Geriatria del Hospital Universitario.

El estudio fue aprobado por el comité de ética local de Niza y solo los participantes con la capacidad de dar su consentimiento para el estudio fueron incluidos. Cada paciente informó su consentimiento antes de la primera evaluación. No fue un estudio al azar de participantes de tres grupos de diagnóstico.

Los datos de video de 49 participantes fueron extraídos de 12 pacientes diagnosticados con la enfermedad del Alzheimer, 23 pacientes diagnosticados con MCI y 14 con controles sanos (Healthy Controls - HC). Todos los diagnósticos fueron realizados por un médico del Hospital Universitario de Geriatria.

Para el grupo formado por los pacientes con la enfermedad de Alzheimer, el diagnóstico fue determinado usando el criterio de diagnóstico propuesto por Dubois, Feldman y col. 2007 que requiere la presencia de un deterioro progresivo de la memoria y pruebas de biomarcadores. Para el grupo que sufre MCI, los pacientes fueron diagnosticados usando el criterio clínico de Petersen (Petersen y col. 1999) y sólo se incluyó a aquellos pacientes con un score mayor a 24 en la examinación del estado mini-mental (MMSE)¹ (M. Folstein, S. Folstein y McHugh 1975). No fueron incluidos aquellos sujetos que con un historial de trauma en la cabeza con pérdida de conocimiento, psicóticos, o con actividad

¹Mini-Mental State Examination

motora aberrante (temblor, rigidez, parkinsonismo) según la sociedad del trastorno del movimiento unificada con la escala de calificación de la enfermedad del Parkinson (Fahn y Elton 1987) con el fin de controlar los posibles trastornos motores que influyen en la capacidad para realizar las actividades instrumentales de la vida diaria. Por otra parte, los participantes con una puntuación MMSE por debajo de 16 se excluyeron a fin de evitar que el participante sufra de experimentar esta evaluación como un gran fracaso.

Cada participante se sometió a una evaluación neuropsicológica estandarizada con un psicólogo. Además, se recogió información médica, clínica y demográfica. El funcionamiento cognitivo global se evaluó mediante el MMSE (M. Folstein, S. Folstein y McHugh 1975) (ver tabla 5.3).

Características	Cantidad de sujetos N = 49	Grupo		
		Healthy Control N = 14	MCI N = 23	AD N = 12
Femenino (%)	26 (53.1%)	9 (64.3%)	10 (43.5%)	7 (57.33%)
Edad, desviación estándar	77,7 ± 7,3	74,1 ± 6,6	77,6 ± 6,2	82 ± 8
MMSE, mean ± SD	25,6 ± 3,1	28,4 ± 1,1	25,5 ± 2,1	22,67 ± 3,6

Healthy Control, control sano; MCI, deterioro cognitivo leve; AD, enfermedad de Alzheimer;
MMSE, examinación del estado mini-mental

Tabla 5.3: Información de los participantes

5.3. Evaluación de la autonomía y predicción de demencia

Utilizando el perfil de comportamiento extraído por el SME, dos modelos *Naïve Bayes* fueron entrenados para clasificar a los participantes en las clases específicas de estado y nivel de autonomía cognitiva de acuerdo a su desempeño en el protocolo clínico. Es importante mencionar que la autonomía se divide en: **Bien**, **Mediocre** y **Mal**; y el diagnóstico de demencias en: **controles sanos**², **MCI**³ y **enfermedad de Alzheimer**⁴.

Para determinar la mejor combinación de parámetros para la clasificación de la autonomía y demencia, se empleó el enfoque *wrapper* (Kohavi y G. John 1997) basado en el método de la búsqueda del mejor primero y el clasificador *Naïve Bayes* (M. A. Hall y Holmes 2003). Tanto la creación de los modelos, como la evaluación del desempeño de los clasificadores, se empleó la técnica *k-fold cross-validation* (con $k = 20$).

Todos los experimentos de clasificación se llevaron a cabo utilizando la plataforma *Weka* (M. Hall y col. 2009). La implementación del clasificador *Naïve Bayes* en *Weka* está basado en el trabajo de G. H. John y Pat Langley 1995.

²Healthy controls - HC

³MCI = Mild Cognitive Impairment. En español, Deterioro Cognitivo Leve

⁴Alzheimer Disease - AD

Selección de características. Wrapper en Weka

El enfoque *wrapper* utilizado para la selección de características es llevado a cabo por la herramienta *Weka* tal como hemos mencionado. Por lo tanto, explicaremos como trabaja internamente para llegar al resultado deseado.

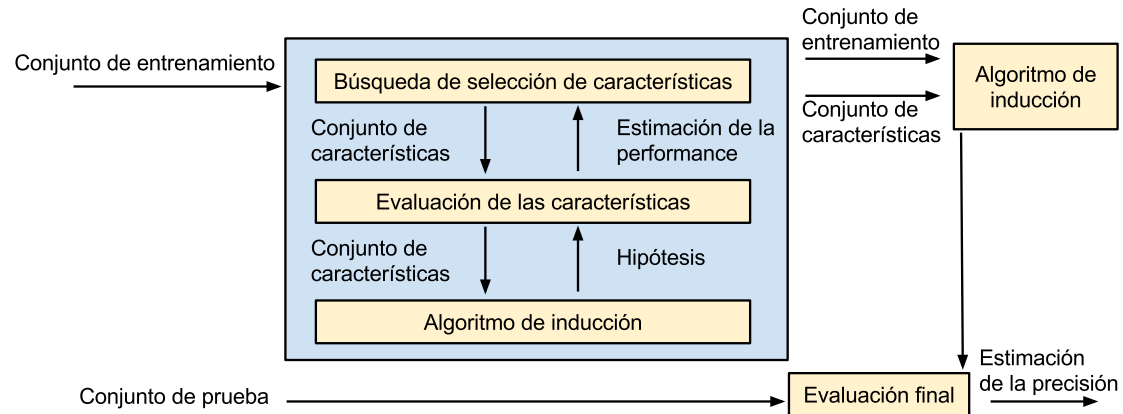


Figura 5.4: Enfoque *wrapper* para la selección de un subconjunto de características. El algoritmo de inducción se utiliza como una “caja negra” por el algoritmo de selección de subconjuntos.

Como se muestra en la Figura 5.4, la selección de un subconjunto de características se realiza utilizando al algoritmo de inducción como una “caja negra” (es decir, no se necesita conocimientos del algoritmo, sólo de la interfaz). El algoritmo de selección de un subconjunto de características lleva a cabo una búsqueda de un buen subconjunto utilizando el propio algoritmo de inducción como parte de la función de evaluación. La precisión de los clasificadores inducidos se estima utilizando técnicas de estimación de la precisión, en este caso *cross-validation* (Ver Sección 4.6).

El enfoque *wrapper* realiza una búsqueda en el espacio de los posibles parámetros. Una búsqueda requiere un espacio de estados, un estado inicial, una condición de terminación dada por el algoritmo *Naïve Bayes*, y un motor de búsqueda, que será “búsqueda del mejor primero” donde la idea es la de seleccionar el nodo más prometedor (en este caso por nodo se refiere a una característica) que se ha generado hasta el momento que todavía no haya sido expandido.

El objetivo de la búsqueda es encontrar el estado con la evaluación más alta, usando una función heurística para guiarla. Dado que se desconoce la precisión real del clasificador inducido, se utiliza la estimación de la precisión tanto de la función heurística como de la función de evaluación. La función de evaluación usada es *5-fold cross-validation* repetida en varias ocasiones. El número de repeticiones es determinado sobre la marcha al ver la desviación estándar de la estimación de la precisión, asumiendo que son independientes. Si la desviación estándar de la estimación de la precisión está por encima del 1 % y no se han ejecutado *5-fold cross-validation*, se ejecuta otro *cross-validation*.

Esta heurística tiene una propiedad que obliga a la estimación de la precisión a ejecutar más veces la técnica *cross-validation* en pequeñas base de datos que en grandes conjuntos de datos. Debido a que pequeñas base de datos requieren menos tiempo para aprender, el tiempo de la estimación de la precisión global, que es el producto del tiempo de ejecución del algoritmo de inducción y el tiempo de *cross-validation*, no crece demasiado rápido.

Por último resta definir el estado inicial de la búsqueda y como se realiza dicha búsqueda. Para ello primero explicaremos brevemente dos términos importantes: *forward selection*, se refiere a la búsqueda que empieza con un conjunto vacío de características y *backward elimination*, el cual empieza la búsqueda con el conjunto total de características. El estado inicial que se utilizó en este trabajo es el conjunto vacío de características, por lo tanto usamos el enfoque *forward selection*. La razón principal se debe a una cuestión de tiempos computacionales; construir clasificadores cuando sólo hay una poca cantidad de características en los datos es más veloz.

Por lo tanto, usando éste método de selección de características brindado por *Weka*, hemos reducido el perfil del comportamiento del paciente a los parámetros más relevantes para la clasificación de la autonomía y la demencia. La selección de características para ambas tareas de clasificación (autonomía y diagnóstico de demencia) empezaron con el conjunto vacío tal como se mencionó anteriormente.

Uso del clasificador *Naïve Bayes*

Hemos elegido el clasificador *Naïve Bayes* debido a su naturaleza probabilística, que cuantifica la pertinencia de la actuación de un participante para cada clase evaluada. Aunque este método asume la independencia condicional entre los parámetros de entrada, una suposición que demuestra ser poco realista para la mayoría de las aplicaciones prácticas, tiende a desempeñarse razonablemente bien comparado con otros métodos más sofisticados, como ser el caso de Support Vector Machine⁵ (G. H. John y Pat Langley 1995; J. Huang, Lu y Ling 2003), con la ventaja de tener un mucho menor tiempo de ejecución y requiriendo muy pocos datos de entrenamiento (Matwin y Sazonova 2012).

La figura 5.5 resume lo mencionado hasta el momento.

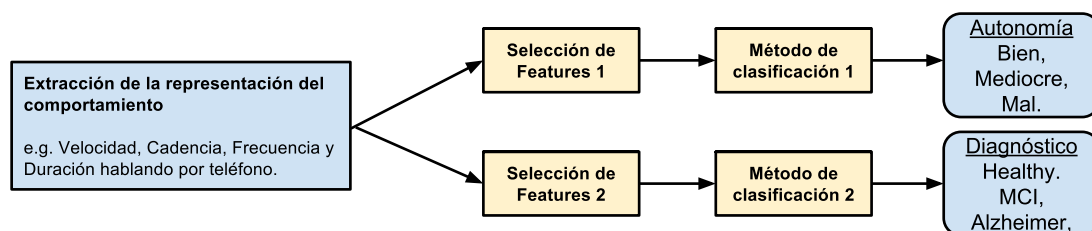


Figura 5.5: Evaluación de la autonomía y clasificación del diagnóstico de demencia

⁵Support Vector Machine = Máquinas de Vectores de Soporte

Es importante entender y resaltar que cuando se utiliza la técnica de *cross-validation* para la evaluación del clasificador, estamos hablando de un *cross-validation* como un bucle exterior independiente al *5-fold cross-validation* repetido que es parte del algoritmo de selección de un subconjunto de características que se explicó en la Sección 5.3. Con esto queremos dejar en claro que el proceso de selección de características es completamente independiente a la evaluación de la clasificación, siendo éstas dos etapas distintas. Primero se realiza el algoritmo de selección de un subconjunto óptimo de características, para luego con dichas características como entrada, evaluar la clasificación. De esta manera, no se establece un sesgo a la hora de realizar la evaluación de la autonomía y predicción del diagnóstico de demencia.

5.4. Validación del sistema de reconocimiento de eventos

La validación del sistema de reconocimiento de eventos se midió en base a los índices de las ecuaciones 5.1 y 5.2 respectivamente.

$$Recall = \frac{TP}{TP + FN} \quad (5.1)$$

$$Precision = \frac{TP}{TP + FP} \quad (5.2)$$

donde, TP: *True Positive rate*, FP: *False Positive rate*, y FN: *False Negative rate*.

La tabla 5.6 muestra los resultados sobre la precisión para la detección de los eventos que se establecieron en el protocolo médico (Actividades guiadas: tarea única y tarea doble. Actividades semi-guiadas: todas las actividades instrumentales de la vida diaria) los cuales fueron anotados por expertos del dominio de manera manual mientras observaban el video experimental.

Actividades guiadas	Recall	Precision
Tarea Simple	100 %	88 %
Tarea Doble	100 %	98 %

Actividades semi-guiadas	Recall	Precision
Uso de mapa de bus	58 %	62.5 %
Preparación de la medicación	87 %	93 %
Regar las plantas	80 %	63 %
Leer un artículo	60 %	88 %
Preparar café	90 %	68 %
Hablar por teléfono	89 %	89 %

Tabla 5.6: Evaluación del sistema. N = 49 pacientes.

Dentro de las actividades guiadas, la tarea simple obtuvo un 88 % de precisión, mientras que la tarea doble un 98 %. En las actividades semi-guiadas (las actividades instrumentales de la vida diaria) , “Preparación de la medicación” fue la que obtuvo el valor más alto de precisión con un 93 %, seguida por “Hablar por teléfono” con un 89 % y “Leer un artículo” con un 88 %.

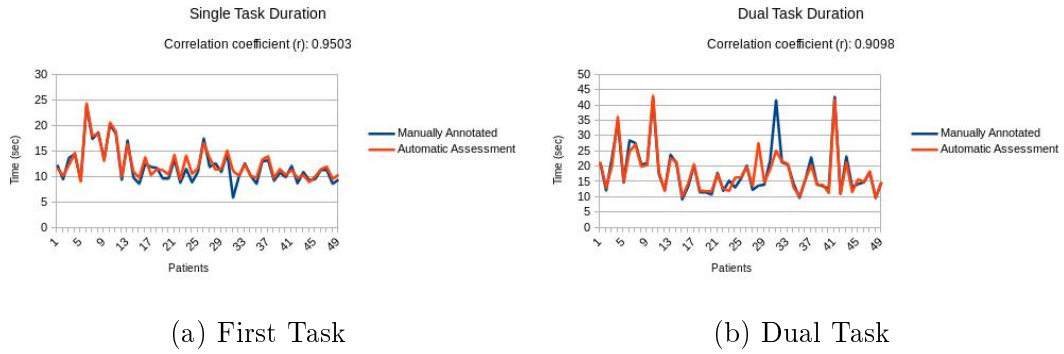


Figura 5.7: Coeficiente de correlación en ambas tareas.

Además se evaluó que tan exitoso es el sistema a la hora de capturar la duración de los eventos en las tareas físicas realizadas por los pacientes en las actividades guiadas. En las Figuras 5.7 se pueden ver graficadas las duraciones de los eventos anotadas por los especialistas (Manually annotated) y la duración capturada por el sistema (Automatic assessment).

El coeficiente de correlación $r \in [-1, 1]$. Mientras más próximo a 1 sea r , mejor es la evaluación en la duración de las tareas. Como podemos ver en Single Task $r = 0,9503$ y en Dual Task $r = 0,9098$. Por lo tanto, nuestro sistema responde de manera exitosa.

5.5. Clasificación de los participantes - Resultados

En la Tabla 5.8 se presenta un resumen con los resultados más importantes de la evaluación de la autonomía y diagnóstico de demencia. El procedimiento de clasificación se basa intrínsecamente en las características extraídos de forma automática de las actividades guiadas y semi-guiadas desempeñadas por el paciente durante el protocolo clínico. Con fines comparativos, también hemos entrenado dos clasificadores basados únicamente en datos de comportamiento sobre las tareas físicas o los datos derivados de las AIVDs. Nuestra hipótesis es que la combinación de la información de los dos escenarios del protocolo aumenta la precisión de la clasificación, ya que proporcionan información diferente pero complementaria sobre la actuación de los participantes en las actividades de la vida diaria, por ejemplo, el rendimiento motor y cognitivo. Para los tres clasificadores, el conjunto de datos es el mismo y contiene en total 49 pacientes. Todas las actividades se detectaron de forma automática con alta sensibilidad y precisión como mostraron los resultados mencionados anteriormente (ver Tabla 5.6).

Performance	Actividades		
	Guiadas	Semi-guiadas	Todas
<i>Evaluación de la autonomía</i>			
Instancias correctamente clasificadas	37 (75.5102 %)	38 (77.551 %)	41 (83.6735 %)
Instancias incorrectamente clasificadas	12 (24.4898 %)	11 (22.449 %)	8 (16.3265 %)
<i>Evaluación del diagnóstico de demencia</i>			
Instancias correctamente clasificadas	36 (73.4694 %)	30 (61.2245 %)	36 (73.4694 %)
Instancias incorrectamente clasificadas	13 (26.5306 %)	19 (38.7755 %)	13 (26.5306 %)

Tabla 5.8: Resultados de la clasificación

En las secciones siguientes se presentarán los resultados con mayores detalles. Para ello mostraremos las características seleccionadas basado en el método de la búsqueda del mejor primero dentro del enfoque wrapper (Kohavi y G. John 1997) y los resultados al aplicar el clasificador *Naïve Bayes* (M. A. Hall y Holmes 2003) para los escenarios por separado y luego para todas las actividades.

Predicción del diagnóstico de demencia

La tabla que se encuentra a continuación es un resumen de los resultados en la predicción del diagnóstico de demencia mientras los pacientes: (a) sólo realizan las actividades guiadas, (b) sólo realizan las actividades semi-guiadas y (c) por último cuando se juntan todas las actividades y se enriquece el perfil de los participantes. Luego tendremos en detalles los resultados de (a), (b) y (c), como así que características fueron seleccionadas con el método nombrado y desarrollado anteriormente (Sección 5.3). Todo esto fue posible gracias a la herramienta WEKA (M. Hall y col. 2009) utilizada tanto para la selección de características como para la clasificación de los pacientes en términos de diagnóstico de demencia.

	Actividades		
	Guiadas	Semi-guiadas	Todas
Instancias correctamente clasificadas	36 (73.4694 %)	30 (61.2245 %)	36 (73.4694 %)
Instancias incorrectamente clasificadas	13 (26.5306 %)	19 (38.7755 %)	13 (26.5306 %)

Cantidad de participantes = 49

Actividades guiadas

Clase	TP Rate	FP Rate	Precision	Recall
Healthy	0.857	0.086	0.8	0.857
MCI	0.739	0.192	0.773	0.739
Alzheimer	0.583	0.135	0.583	0.583

Tabla 5.9: Detalle de precisión en actividades guiadas en la predicción del diagnóstico de demencia

Matriz de confusión				
a	b	c	Clasificado como	
12	1	1	a = Healthy	1 paciente fue clasificado como MCI cuando era Healthy. 1 paciente fue clasificado como ALZHEIMER cuando era HEALTHY.
2	17	4	b = MCI	2 pacientes fueron clasificados como HEALTHY cuando eran MCI. 4 pacientes fueron clasificados como ALZHEIMER cuando eran MCI.
1	4	7	c = Alzheimer	1 paciente fue clasificado como HEALTHY cuando era ALZHEIMER. 4 pacientes fueron clasificados como MCI cuando eran ALZHEIMER.

Las características seleccionadas

- Age,
- Single_Task Total Duration,
- Single_Task Average Steps,
- Single_Task Standard Deviation of Steps,
- Single_Task Max Strides⁶,
- Single_Task Speed Average from Centroid Information⁷,
- Double_Task Cadence⁸,
- Double_Task Max Steps,
- Double_Task Min Steps,

⁶Strides = dos pasos consecutivos

⁷la velocidad se obtiene desde el framework de seguimiento de objetos

⁸Cantidad de pasos por minuto

Actividades semi-guiadas

Clase	TP Rate	FP Rate	Precision	Recall
Healthy	0.571	0.171	0.571	0.571
MCI	0.783	0.423	0.621	0.783
Alzheimer	0.333	0.054	0.667	0.333

Tabla 5.10: Detalle de precisión en las actividades semi-guiadas en la predicción del diagnóstico de demencia

Matriz de confusión				
a	b	c	Clasificado como	
8	5	1	a = Healthy	5 pacientes fueron clasificados como MCI cuando eran HEALTHY. 1 paciente fue clasificado como ALZHEIMER cuando era HEALTHY.
4	18	1	b = MCI	4 pacientes fueron clasificados como HEALTHY cuando eran MCI. 1 paciente fue clasificado como ALZHEIMER cuando era MCI.
2	6	4	c = Alzheimer	2 pacientes fueron clasificados como HEALTHY cuando eran ALZHEIMER. 6 pacientes fueron clasificados como MCI cuando eran ALZHEIMER.

Las características seleccionadas

- Age,
- Sex,
- Person_reading_inChairReadingTable Duration (frames),
- Person_reading_inChairReadingTable Latency (seconds),
- Person_making_Tea_Frequency (times),
- Person_using_Phone_Frequency Healthy Probability⁹.

⁹Parámetro gaussiano deducido.

Todas las actividades

Clase	TP Rate	FP Rate	Precision	Recall
Healthy	0.714	0.083	0.769	0.714
MCI	0.826	0.192	0.792	0.826
Alzheimer	0.583	0.135	0.583	0.583

Tabla 5.11: Detalle de precisión en todas las actividades en la predicción del diagnóstico de demencia

Matriz de confusión				
a	b	c	Clasificado como	
10	2	2	a = Healthy	2 pacientes fueron clasificados como MCI cuando eran HEALTHY. 2 pacientes fueron clasificados como ALZHEIMER cuando eran HEALTHY.
1	19	3	b = MCI	1 paciente fue clasificado como HEALTHY cuando era MCI. 3 pacientes fueron clasificados como ALZHEIMER cuando eran MCI.
2	3	7	c = Alzheimer	2 pacientes fueron clasificados como HEALTHY cuando eran ALZHEIMER. 3 pacientes fueron clasificados como MCI cuando eran ALZHEIMER.

Las características seleccionadas

- Age,
- Single_Task Average Steps,
- Single_Task Speed Average from Centroid Information¹⁰,
- Dual_Task Max Steps,
- Dual_Task Min Steps,
- Person_reading_inChairReadingTable Duration (Frames).

¹⁰la velocidad se obtiene desde el framework de seguimiento de objetos

Predicción de la autonomía

La tabla que se encuentra a continuación es un resumen de los resultados de la evaluación de la autonomía de los pacientes mientras: (a) sólo realizan las actividades guiadas, (b) sólo realizan las actividades semi-guiadas y (c) por último cuando se juntan todas las actividades y se enriquece el perfil de los participantes. Luego tendremos en detalles los resultados de (a), (b) y (c), como así que características fueron seleccionadas con el método nombrado y desarrollado anteriormente (Sección 5.3).

	Actividades		
	Guiadas	Semi-guiadas	Todas
Instancias correctamente clasificadas	37 (75.5102 %)	38 (77.551 %)	41 (83.6735 %)
Instancias incorrectamente clasificadas	12 (24.4898 %)	11 (22.449 %)	8 (16.3265 %)

Cantidad de participantes = 49

Actividades guiadas

Clase	TP Rate	FP Rate	Precision	Recall
Bien	0.826	0.231	0.76	0.826
Mediocre	0.625	0.091	0.769	0.625
Mal	0.8	0.077	0.727	0.8

Tabla 5.12: Detalle de precisión en las actividades guiadas en la predicción de la autonomía

Matriz de confusión				
a	b	c	Clasificado como	
19	3	1	a = Bien	3 pacientes fueron clasificados como MEDIOCRE cuando eran BIEN. 1 paciente fue clasificado como MEDIOCRE cuando era BIEN.
4	10	2	b = Mediocre	4 pacientes fueron clasificados como BIEN cuando eran MEDIOCRE. 2 pacientes fueron clasificados como MAL cuando eran MEDIOCRE.
2	0	8	c = Mal	2 pacientes fueron clasificados como BIEN cuando eran MAL.

Las características seleccionadas

- Single_Task Gap Duration¹¹,
- Dual_Task Distance,
- Dual_Task Gap Duration,
- Dual_Task Number of Strides,
- Double_Task Max Steps.

¹¹Tiempo entre que el participante llega a la zona de stop y empieza a caminar hasta la zona de conteo nuevamente.

Actividades semi-guiadas

Class	TP Rate	FP Rate	Precision	Recall
Bien	0.957	0.192	0.815	0.957
Mediocre	0.688	0.121	0.733	0.688
Mal	0.6	0.026	0.857	0.6

Tabla 5.13: Detalle de precisión en las actividades semi-guiadas en la predicción de la autonomía

Matriz de confusión				
a	b	c	Clasificado como	
22	1	0	a = Bien	1 paciente fue clasificado como MEDIOCRE cuando era BIEN.
4	11	1	b = Mediocre	4 pacientes fueron clasificados como BIEN cuando eran MEDIOCRE. 1 paciente fue clasificado como MAL cuando era MEDIOCRE.
1	3	6	c = Mal	1 paciente fue clasificado como BIEN cuando era MAL. 3 pacientes fueron clasificados como MEDIOCRE cuando eran MAL.

Las características seleccionadas

- Person_using_PharmacyBasket Frequency (times),
- Person_using_PharmacyBasket_Frequency Alzheimer Probability,
- Person_using_PharmacyBasket Duration (seconds),
- Person_making_Tea Frequency (times),
- Person_using_Phone Expected¹²,
- Person_using_Phone_Frequency MCI Probability,
- Person_using_Phone Latency (Seconds).

¹²número de veces que el paciente debe hacer la actividad, menos las veces que lo hizo.

Todas las actividades

Class	TP Rate	FP Rate	Precision	Recall
Bien	0.87	0.77	0.909	0.87
Mediocre	0.813	0.91	0.813	0.813
Mal	0.8	0.77	0.727	0.8

Tabla 5.14: Detalle de precisión en todas las actividades en la predicción de la autonomía

Matriz de confusión				
a	b	c	Clasificado como	
20	2	1	a = Bien	2 pacientes fueron clasificados como MEDIOCRE cuando eran BIEN. 1 paciente fue clasificado como MAL cuando era BIEN.
1	13	2	b = Mediocre	1 paciente fue clasificado como BIEN cuando era MEDIOCRE. 2 pacientes fueron clasificados como MAL cuando eran MEDIOCRE.
1	1	8	c = Mal	1 paciente fue clasificado como BIEN cuando era MAL. 1 paciente fue clasificado como MEDIOCRE cuando era MAL.

Las características seleccionadas

- Single_Task Total Duration,
- Single_Task Gap Duration,
- Single_Task Standard Deviation Steps,
- Dual_Task Gap Duration.
- Dual_Task Max Steps,
- Person_using_PharmacyBasket Frequency (times),
- Person_using_PharmacyBasket Duration (seconds).

Distintas técnicas validación y entrenamiento

Hasta ahora, todos los resultados presentados fueron a partir de la utilización del método *cross-validation*. De manera que este trabajo de investigación tenga una validación extra por fuera de técnicas implementadas previamente, obtendremos nuevos resultados dividiendo nuestra base de datos manualmente y haciendo un promedio de todos esos resultados conseguidos. Por lo tanto, se decidió separar los datos en un número razonablemente chico para contrarrestar con los *20-fold cross-validation*. Entonces, se formaron 3 grupos tratando de mantener un balance en la cantidad de pacientes por clase a clasificar. Al primer grupo lo hemos tomado como datos de entrenamiento, mientras que a la conjunción de los otros 2 restantes como datos de validación o testeo. Luego el segundo grupo fue tomado como datos de entrenamiento y el resto como datos de validación; y por último lo mismo con el tercer grupo. Todas estas combinaciones están presentadas en la tabla 5.15.

	Autonomía	Diagnóstico de demencia
Primer grupo con 20-fold cross-validation	69.69 %	60.60 %
Primer grupo como datos de prueba	81.25 %	81.25 %
Segundo grupo con 20-fold cross-validation	78.78 %	66.67 %
Segundo grupo como datos de prueba	87.50 %	56.25 %
Tercer grupo con 20-fold cross-validation	81.25 %	59.38 %
Tercer grupo como datos de prueba	64.70 %	64.70 %
Promedio con 20-fold cross-validation	76.57 %	62.22 %
Promedio de los grupos como datos de prueba	77.82 %	67.40 %
Todas las actividades con 20-fold cross-validation	83.67 %	73.46 %

Tabla 5.15: Tabla de comparación entre nuestra técnica de evaluación y *stratified k-fold cross-validation*

Si bien los resultados finales obtenidos con la división manual de los datos no llegan a alcanzar un valor tan alto como los conseguidos con la técnica de *cross-validation*, el hecho de que esa diferencia no sea abultada, nos permite llegar a la conclusión de que el trabajo estuvo bien encaminado y además que, la técnica de evaluación *stratified cross-validation* es la adecuada para la clasificación y evaluación en este trabajo.

Capítulo 6

Conclusión

6.1. Discusión

En el presente trabajo de investigación podemos concluir que es posible la evaluación en tiempo real y con alta precisión de la autonomía de los pacientes mediante el desempeño de las actividades instrumentales de la vida diaria con la ayuda de un sistema de monitoreo de eventos que simplemente se basa en la extracción de información a través de una cámara RGB-D, un detector de personas, un seguimiento de objetos, *feet detection* y una ontología basada en restricciones para modelar actividades y detectarlas. Los resultados obtenidos no sólo son significativamente altos para una correcta evaluación de la autonomía, sino también para el estado cognitivo en términos de diagnóstico. Esto significa, que el sistema propuesto puede convertirse en una herramienta muy útil que proporciona a los médicos un diagnóstico con información relevante y así poder mejorar la evaluación de la autonomía en los pacientes, que sufren de la enfermedad de Alzheimer o de un deterioro cognitivo leve, en tiempo real disminuyendo los sesgos de investigación¹

Los resultados demuestran además que el análisis de la marcha sumado a la evaluación de las actividades de la vida diaria, pueden proporcionar una metodología fiable y precisa para evaluar el desempeño de los pacientes en la vida diaria, que podría ser utilizado tanto a nivel de diagnóstico. como a nivel de rehabilitación. Cuando se trabajó con las características de ambos escenarios la clasificación de la autonomía llegó a casi un 84 % mientras que la predicción del diagnóstico de demencia alcanzó alrededor del 74 % (aunque en este caso sólo con las actividades físicas se logró el mismo resultado final que con todas las actividades pero con distintas precisiones en cada clase). Si bien hubiésemos esperado obtener, en un principio, casi el mismo porcentaje en ambas predicciones ya que existe una relación entre la autonomía y el diagnóstico (relaciones: “Bien” \mapsto “Healthy”; “Mediocre” \mapsto “MCI”; “Mal” \mapsto “Alzheimer”), el comportamiento impredecible de los pacientes enfermos sumado a que nuestro sistema no tiene una precisión del 100 % en los eventos a detectar, provocaron esa caída de 10 % en el resultado.

¹El sesgo de investigación constituye un proceso en el cual los científicos que realizan la investigación influyen en los resultados, con el fin de representar un determinado resultado.

Por lo tanto, lo que se logró es encontrar una evaluación CUANTITATIVA en cuanto a la autonomía y la predicción del diagnóstico de demencia con la utilización de un sistema de monitoreo de video, en lugar de una evaluación cualitativa tradicionalmente confiada en escalas y cuestionarios que no siempre son sensibles a los primeros cambios funcionales.

El principal interés de este estudio fue demostrar la aplicación práctica de la utilización de un sistema de monitoreo de vídeo en la práctica clínica. Ahora, una vez que el uso del sistema ha sido validado por una correlación significativa con las puntuaciones neuropsicológicas de la prueba, en particular para el funcionamiento ejecutivo, y con niveles de detección de alta precisión, puede ser empleado como una herramienta de evaluación de apoyo dentro de chequeos clínicos de rutina también a nivel de rehabilitación e incluso pasar a ser utilizado en entornos específicos como hogares para ancianos.

6.2. Contribuciones a futuro

Como en la mayoría de los trabajos de investigación, existen muchas cosas por mejorar para que los resultados obtenidos sean aún mejores. Una de los puntos más importantes sería lograr una base de datos de pacientes mucho mayor que tan sólo 49 y que sea más equilibrada con respecto a las clases a evaluar. De esa manera, los datos de entrenamiento del modelo podrían ajustarse mejor a nuestras necesidades.

Si bien el detector de personas se desempeñó con éxito, algunos errores en el mismo provocaron la extracción de datos erróneos, por lo que una mejora en el algoritmo de eliminación de ruido es un punto a mejorar.

Por último, para que la detección de eventos logre una precisión más alta, se está trabajando en el agregado de cámaras sobre el hombro del participante que ejecuten un algoritmo de detección de objetos. De esa manera, el sistema de monitoreo de eventos dejará de confundir cuando el paciente se encuentre en una zona comúnmente utilizada para determinada acción pero realizando otra (e.g., dentro de la zona de hablar por teléfono, el paciente se encuentra divagando).

Bibliografía

- Allen, James F. (1983). «Maintaining knowledge about temporal intervals». En: *Communications of the ACM*.
- Arrighi, H.M. y col. (2005). «Prevalence and severity of gait disorders in Alzheimer's and non-Alzheimer's dementias.» En: *J. Am. Geriatr. Soc.* 53, págs. 1681-1687.
- Baddeley, A. y col. (1986). «Dementia and working memory.» En: *Q. J. Exp. Psychol.* A 38, págs. 603-618.
- Bak, S. y col. (2010). «Person Re-identification Using Spatial Covariance Regions of Human Body Parts.» En: *AVSS*.
- Barnich, O. y M. V. Droogenbroeck (2011). «Vibe: A universal background subtraction algorithm for video sequences». En: *IEEE Transactions on Image Processing*.
- Chau, Duc Phu y col. (2014). «Automatic Tracker Selection w.r.t Object Detection Performance». En: *IEEE Winter Conference on Applications of Computer Vision (WACV 2014)*.
- Clare, L. y col. (2003). «Cognitive rehabilitation and cognitive training for early-stage Alzheimer's disease and vascular dementia. [Review]». En: *Cochrane Database Syst. Rev.* 4.
- Cleary, J. G., S. Legg e I. H. Witten (1996). «An MDL estimate of the significance of rules». En: *Information, Statistics, and Induction in Science*.
- Cotelli, M., M. Calabria y O. Zanetti (2006). «Cognitive rehabilitation in Alzheimer's disease. [Review]». En: *Aging Clin. Exp. Res.* 18, págs. 141-143.
- Crispim, Carlos Fernando y col. (2013). «Evaluation of a Monitoring System for Event Recognition of Older People». En: *10th IEEE International Conference on Advanced Video and Signal Based Surveillance*, págs. 165-170.
- Dalal, N. y B. Triggs (2005). «Histograms of oriented gradients for human detection». En: *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*. Vol. 1, 886-893 vol. 1. DOI: [10.1109/CVPR.2005.177](https://doi.org/10.1109/CVPR.2005.177).
- DeBettignies, B. H., R. K. Mahurin y F. J. Pirozzolo (1990). «Insight for impairment in independent living skills in Alzheimer's disease and multi-infarct dementia. [Research Support, Non-U.S. Gov't Research Support, U.S. Gov't, P.H.S.]» En: *J. Clin. Exp. Neuropsychol* 12, págs. 355-363.
- Domingos, P. y M. Pazzani (1996). «Beyond Independence: Conditions for the Optimality of the Simple Bayesian Classifier». En: *Machine Learning: Proceedings of the Thirteenth International Conference on Machine Learning*.

- Dubois, B., H.H. Feldman y col. (2007). «Research criteria for the diagnosis of Alzheimer's disease: revising the NINCDS-ADRDA criteria». En: *The Lancet. Neurology* 6(8), págs. 734-46.
- Dubois, B., A. Slachevsky y col. (2000). «The FAB: a frontal assessment battery at bedside. [Comment Research Support, Non-U.S. Gov't].» En: *Neurology* 55, págs. 1621-1626.
- Elgammal, A., D. Harwood y L. Davis (2000). «Non-parametric model for background subtraction». En: *IEEE Frame-Rate Workshop*.
- Fahn, S. y R.L. Elton (1987). «UPDRS program members. Unified Parkinsons disease rating scale». En: *Recent Developments in Parkinsons Disease*, eds S. Fahn, M. Goldstein, and D. B. Calne (Florham Park, NJ), págs. 153-163.
- Folstein, M., S.E. Folstein y P.R. McHugh (1975). «“Mini-mental state”: a practical method for grading the cognitive state of patients for the clinician». En: *Journal of psychiatric research* 12(3), 89:98.
- Fosty, Baptiste, Gregory Ben Sadoun y Francois Bremond (2015). «Computation of the walking speed of a person on a treadmill using an RGB-D camera». En: *Unpublished*.
- Gamberger, Dragan y Nada Lavrac (1997). «Conditions for Occam's Razor Applicability and Noise Elimination». En: *Ninth European Conference on Machine Learning*.
- Geisser, S. (1975). «The predictive sample reuse method with applications». En: *Journal of the American Statistical Association* 70(350), págs. 320-328.
- Gold, D. A. (2012). «An examination of instrumental activities of daily living assessment in older adults and mild cognitive impairment. [Research Support, Non-U.S. Gov't Review]». En: *J. Clin. Exp. Neuropsychol* 34, págs. 11-34.
- Hall, M. A. y G. Holmes (2003). «Benchmarking attribute selection techniques for discrete class data mining». En: *IEEE Transactions on Knowledge and Data Engineering* 15, págs. 1437-1447.
- Hall, Mark A. (1999). «Correlation-based Feature Selection for Machine Learning». Tesis doct. The university of Waikato.
- Hall, M. y col. (2009). «The WEKA data mining software: an update». En: *ACM SIGKDD Explorations Newsletter table of contents archive. Sponsor SIGKDD ACM Special Interest Group on Knowledge Discovery in Data*. eds M. A. Hall and G. Holmes (New York, NY: ACM) 11, págs. 10-18.
- Hofmann, M., P. Tiefenbacher y G. Rigoll (2012). «Background segmentation with feedback: The pixel-based adaptive segmenter». En: *Computer Vision and Pattern Recognition Workshops*.
- Huang, J., J. Lu y C.X. Ling (2003). «Comparing naive Bayes, decision trees, and SVM with AUC and accuracy». En: *Third IEEE International Conference on Data Mining ICDM*, págs. 553-556.
- John, George H. y Pat Langley (1995). «Estimating Continuous Distributions in Bayesian Classifiers». En: *Eleven Conference on Uncertainty in Artificial Intelligence*, Morgan Kaufmann Publishers, San Mateo.
- Karzmark, P. y col. (2012). «Comparison of the frontal systems behavior scale and neuropsychological tests of executive functioning in predicting instrumental activities of daily living». En: *Appl. Neuropsychol. Adult* 19, págs. 81-85.

- Kohavi, R. (1995). «Wrappers for Performance Enhancement and Oblivious Decision Graphs». Tesis doct. Stanford University.
- Kohavi, R. y G. John (1997). «Wrappers for feature selection». En: *Artificial Intelligence* 97(1-2), págs. 273-324.
- König, Alexandra y col. (2015). «Ecological assessment of autonomy in instrumental activities of daily living in dementia patients by the means of an automatic video monitoring system». En: *Frontier in Aging Neuroscience* 7. Article 98.
- König, A. y col. (2014). «Validation of an automatic video monitoring system for the detection of instrumental activities of daily living in dementia patients». En: *Journal of Alzheimer's Disease* 44, págs. 675-685.
- Kononenko, I. e I. Bratko. (1993). «Machine Learning». En: cap. Information-based evaluation criterion for classifier's performance, 6:67-80.
- Kuhn, Harold W. (1955). «The Hungarian Method for the assignment problem». En: *Naval Research Logistic Quarterly* 2, págs. 83-97.
- Kuo, C., C. Huang y R. Nevatia (2010). «Multi-target tracking by online learned discriminative appearance models». En: *CVPR*.
- Langley, P. y S. Sage (1994). «Induction of Selective Bayesian Classifiers». En: *Tenth Conference on Uncertainty in Artificial Intelligence, Seattle, W.A.*
- Lawton, M. P. y E. M. Brody (1969). «Assessment of older people: self-maintaining and instrumental activities of daily living». En: *Gerontologist* 9, págs. 179-186.
- Maquet, D. y col. (2010). «Gait analysis in elderly adult patients with mild cognitive impairment and patients with mild Alzheimer's disease: simple versus dual task: a preliminary report». En: *Clin Physiol Funct Imaging* 30(1), págs. 51-6.
- Marshall, G.A. y col. (2006). «Neuropathologic correlates of activities of daily living in Alzheimer disease. [Research Support, N.I.H., Extramural]». En: *Alzheimer Dis. Assoc. Disord* 20, págs. 56-59.
- Matwin, S. y V. Sazonova (2012). «Direct comparison between support vector machine and multinomial naive bayes algorithms for medical abstract classification». En: *Journal of the American Medical Informatics Association : JAMIA* 19(5), pág. 917.
- Meré, Julio Molleda (2008). «Técnicas de Visión por Computador para la Reconstrucción en Tiempo Real de la Forma 3D de Productos Laminados». Tesis doct. Universidad de Oviedo - Departamento de Informática. Gijón.
- Mitchell, Tom y McGraw Hill (1997). *Machine Learning*.
- Moore, D.J. y col. (2007). «A review of performance-based measures of functional living skills. [Research Support, N.I.H., Extramural Research Support, U.S. Gov't, Non-P.H.S.Review]». En: *J. Psychiatr. Res* 41, págs. 97-118.
- Muir, Susan W. y col. (2012). «Gait assessment in mild cognitive impairment and Alzheimer's disease: The effect of dual-task challenges across the cognitive spectrum». En: *Gait and Posture* 35, págs. 96-100.
- Nadkarni, Neelesh K. y col. (2009). «Spatial and temporal gait parameters in Alzheimer's disease and aging». En: *Gait Posture* 30(4), págs. 452-4.

- Nghiem, A., E. Auvinet y J. Meunier (2012). «Head detection using kinect camera and its application to fall detection». En: *11th International Conference on Information Sciences, Signal Processing and their Applications*.
- Nghiem, Anh-Tuan y Francois Bremond (2014). «Background subtraction in people detection framework for RGB-D cameras». En: *AVSS*.
- Petersen, R. C. y col. (1999). «Mild cognitive impairment: clinical characterization and outcome». En: *Arch. Neurol* 56, págs. 303-308.
- Quinlan, J.R. (1986). *Machine Learning 1: 81-106, Induction of Decision Trees*. Kluwer Academic Publishers, Boston.
- (1993). *C4.5: Programs for Machine Learning by J. Ross Quinlan*. Morgan Kaufmann Publishers, Inc. Kluwer Academic Publishers.
- Razani, J. y col. (2007). «Relationship between executive functioning and activities of daily living in patients with relatively mild dementia. [Research Support, N.I.H., Extramural]». En: *Appl. Neuropsychol* 14, págs. 208-214.
- Reppermund, S. y col. (2013). «Impairment in instrumental activities of daily living with high cognitive demand is an early marker of mild cognitive impairment: the Sydney memory and ageing study». En: *Psychological medicine* 43(11), págs. 2437-45.
- Robert, P.H. y col. (2013). «Recommendations for ICT use in Alzheimer's disease assessment: Monaco CTAD Expert Meeting». En: *J Nutr Health Aging* 17(8), págs. 653-60.
- Romdhane, R. y col. (2012). «Automatic video monitoring system for assessment of Alzheimer's disease symptoms». En: *Health Aging* 16, págs. 213-218.
- Sacco, G. y col. (2013). «Detection of activities of daily living impairment in Alzheimer's disease and mild cognitive impairment using information and communication technology». En: *Clinical Interventions in Aging* 7, págs. 539-549.
- Santner, J. y col. (2010). «Prost: Parallel robust online simple tracking». En: *CVPR*.
- Schaffer, C. (1993). «Selecting a classification method by cross-validation». En: *Machine Learning* 13, págs. 135-143.
- Shi, J. y C. Tomasi (1994). «Good features to track». En: *CVPR*.
- Shotton, J. y A. Fitzgibbon (2011). «Real-time human pose recognition in parts from single depth images». En: *IEEE Conference on Computer Vision and Pattern Recognition*.
- Sikkes, S.A., E.S. de Lange-de Klerk, Y.A. Pijnenburg, F. Gillissen y col. (2012). «A new informant-based questionnaire for instrumental activities of daily living in dementia». En: *Alzheimers Dement* 8(6), págs. 536-43.
- Sikkes, S.A., E.S. de Lange-de Klerk, Y.A. Pijnenburg, P. Scheltens y col. (2009). «A systematic review of instrumental activities of daily living scales in dementia: room for improvement. [Research Support, Non-U.S. Gov't Review]». En: *J. Neurol. Neurosurg. Psychiatr.* 80, págs. 7-12.
- Snyder, P.J. y col. (2014). «Assessing cognition and function in Alzheimer's disease clinical trials: do we have the right tools?». En: *Alzheimers Dement* 10(6), págs. 853-60.
- Spinello, L. y K. O. Arras (2011). «People detection in rgb-d data». En: *International Conference on Intelligent Robots and Systems*.

- Stauffer, C. y W. Grimson (1999). «Adaptive background mixture models for real-time tracking». En: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*.
- Stern, Y. y col. (1990). «Measurement and prediction of functional capacity in Alzheimer's disease». En: *Neurology* 40, págs. 8-14.
- T. Vu, F. Brémond y M. Thonnat (2003). «Automatic Video Interpretation: A Novel Algorithm for Temporal Scenario Recognition». En: *The Eighteenth International Joint Conference on Artificial Intelligence*.
- Tombaugh, T.N. (2004). «Trail making test A and B: normative data stratified by age and education». En: *Arch. Clin. Neuropsychol* 19, págs. 203-214.
- Yang, N.C. y col. (2008). «A fast mpeg-7 dominant color extraction with new similarity measure for image retrieval». En: *J. Visual Communication and Image Representation*.
- Yoon, J. H., D. Y. Kim y K. J. Yoon (2012). «Visual tracking via adaptive tracker selection with multiple features». En: *ECCV*.